# ESP32-C6 Series

## Datasheet

Ultra-low-power SoC with RISC-V single-core microprocessor
2.4 GHz Wi-Fi 6 (802.11ax), Bluetooth® 5 (LE), Zigbee and Thread (802.15.4)
Optional 4 MB flash in the chip's package
30 or 22 GPIOs, rich set of peripherals
QFN40 (5×5 mm) or QFN32 (5×5 mm) package

### Including:

ESP32-C6

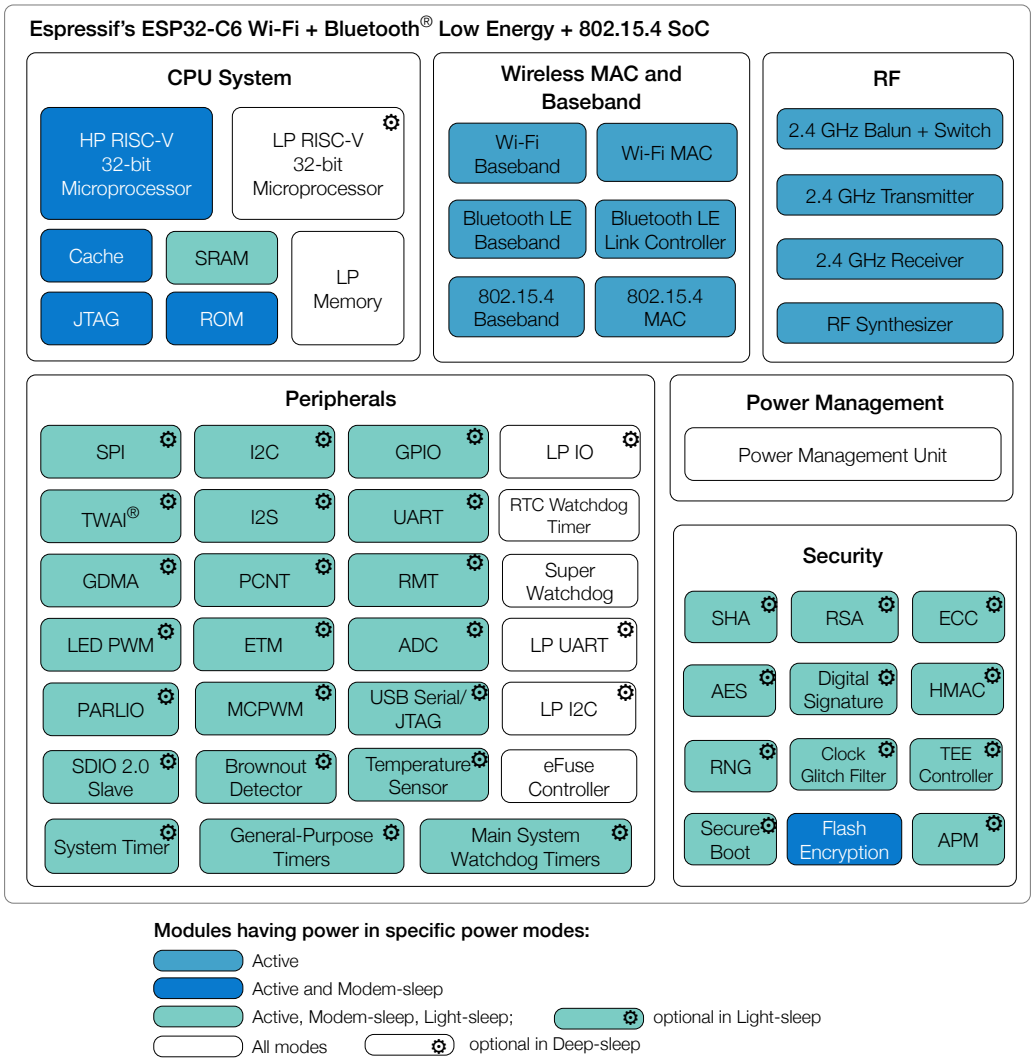
ESP32-C6FH4

Version 1.1
Espressif Systems

# Product Overview

The ESP32-C6 SoC (System on Chip) supports Wi-Fi 6 in 2.4 GHz band, Bluetooth 5, Zigbee 3.0 and Thread 1.3. It consists of a high-performance (HP) 32-bit RISC-V processor, an low-power (LP) 32-bit RISC-V processor, wireless baseband and MAC (Wi-Fi, Bluetooth LE, and 802.15.4), RF module, and numerous peripherals. Wi-Fi, Bluetooth and 802.15.4 coexist with each other and share the same antenna.

The functional block diagram of the SoC is shown below.




**ESP32-C6 Functional Block Diagram**

For more information on power consumption, see Section 4.1.3.7 *Power Management Unit*.

# Features

## Wi-Fi

- 1T1R in 2.4 GHz band
- Operating frequency: 2412 ~ 2484 MHz
- IEEE 802.11ax-compliant
  - 20 MHz-only non-AP mode
  - MCS0 ~MCS9
  - Uplink and downlink OFDMA, especially suitable for simultaneous connections in high-density environments
  - Downlink MU-MIMO (multi-user, multiple input, multiple output) to increase network capacity
  - Beamformee that improves signal quality
  - Channel quality indication (CQI)
  - DCM (dual carrier modulation) to improve link robustness
  - Spatial reuse to maximize parallel transmissions
  - Target wake time (TWT) that optimizes power saving mechanisms
- Fully compatible with IEEE 802.11b/g/n protocol
  - 20 MHz and 40 MHz bandwidth
  - Data rate up to 150 Mbps
  - Wi-Fi Multimedia (WMM)
  - TX/RX A-MPDU, TX/RX A-MSDU
  - Immediate Block ACK
  - Fragmentation and defragmentation
  - Transmit opportunity (TXOP)
  - Automatic Beacon monitoring (hardware TSF)
  - Four virtual Wi-Fi interfaces
  - Simultaneous support for Infrastructure BSS in Station mode, SoftAP mode, Station + SoftAP mode, and promiscuous mode

    *Note that when ESP32-C6 scans in Station mode, the SoftAP channel will change along with the Station channel*
  - Antenna diversity
  - 802.11mc FTM

## Bluetooth®

- Bluetooth LE: Bluetooth 5.3 certified
- Bluetooth mesh
- High power mode (20 dBm)
- Speed: 125 Kbps, 500 Kbps, 1 Mbps, 2 Mbps
- Advertising extensions
- Multiple advertisement sets
- Channel selection algorithm #2
- LE power control
- Internal co-existence mechanism between Wi-Fi and Bluetooth to share the same antenna

## IEEE 802.15.4

- Compliant with IEEE 802.15.4-2015 protocol
- OQPSK PHY in 2.4 GHz band
- Data rate: 250 Kbps
- Thread 1.3
- Zigbee 3.0

## CPU and Memory

- HP RISC-V processor:
  - Clock speed: up to 160 MHz
  - Four stage pipeline
  - CoreMark® score: 464.36 CoreMark；2.90 CoreMark/MHz (160 MHz)
- LP RISC-V processor:
  - Clock speed: up to 20 MHz
  - Two stage pipeline

- L1 cache: 32 KB
- ROM: 320 KB
- HP SRAM: 512 KB
- LP SRAM: 16 KB
- Supported SPI protocols: SPI, Dual SPI, Quad SPI, QPI interfaces that allow connection to flash and other SPI devices off the chip's package
- Flash controller with cache is supported
- Flash in-Circuit Programming (ICP) is supported

## Advanced Peripheral Interfaces

- 30 GPIOs (QFN40), or 22 GPIOs (QFN32)
  - 5 strapping GPIOs
  - 6 GPIOs needed for in-package flash
- Analog interfaces:
  - 12-bit SAR ADC, up to 7 channels
  - Temperature sensor
- Digital interfaces:
  - Two UARTs
  - Low-power (LP) UART
  - Two SPI ports for communication with flash
  - General purpose SPI port
  - I2C
  - Low-power (LP) I2C
  - I2S
  - Pulse count controller
  - USB Serial/JTAG controller
  - Two TWAI® controllers, compatible with ISO 11898-1 (CAN Specification 2.0)
  - SDIO 2.0 slave controller
  - LED PWM controller, up to 6 channels
  - Motor Control PWM (MCPWM)
  - Remote control peripheral (TX/RX)
  - Parallel IO interface (PARLIO)
  - General DMA controller, with 3 transmit channels and 3 receive channels
  - Event task matrix (ETM)
- Timers:
  - 52-bit system timer
  - Two 54-bit general-purpose timers
  - Three digital watchdog timers
  - Analog watchdog timer

## Power Management

- Fine-resolution power control through a selection of clock frequency, duty cycle, Wi-Fi operating modes, and individual power control of internal components
- Four power modes designed for typical scenarios: Active, Modem-sleep, Light-sleep, Deep-sleep
- Power consumption in Deep-sleep mode is 7 $\mu$A
- Low-power (LP) memory remains powered on in Deep-sleep mode

## Security

- Secure boot - permission control on accessing internal and external memory
- Flash encryption - memory encryption and decryption
- 4096-bit OTP, up to 1792 bits for users
- Trusted execution environment (TEE) controller and access permission management (APM)
- Cryptographic hardware acceleration:
  - AES-128/256 (FIPS PUB 197)
  - ECC
  - HMAC
  - RSA
  - SHA (FIPS PUB 180-4)
  - Digital signature

- External Memory Encryption and Decryption (XTS_AES)
- Random Number Generator (RNG)

### RF Module

- Antenna switches, RF balun, power amplifier, low-noise receive amplifier
- Up to +21 dBm of power for an 802.11b transmission
- Up to +19.5 dBm of power for an 802.11ax transmission
- Up to -106 dBm of sensitivity for Bluetooth LE receiver (125 Kbps)

## Applications

With low power consumption, ESP32-C6 is an ideal choice for IoT devices in the following areas:

- Smart Home
- Industrial Automation
- Health Care
- Consumer Electronics
- Smart Agriculture
- POS Machines
- Service Robot
- Audio Devices
- Generic Low-power IoT Sensor Hubs
- Generic Low-power IoT Data Loggers

**Note:**

Check the link or the QR code to make sure that you use the latest version of this document:
https://www.espressif.com/documentation/esp32-c6_datasheet_en.pdf

# Contents

# List of Tables

# List of Figures

# 1 ESP32-C6 Series Comparison

## 1.1 Nomenclature

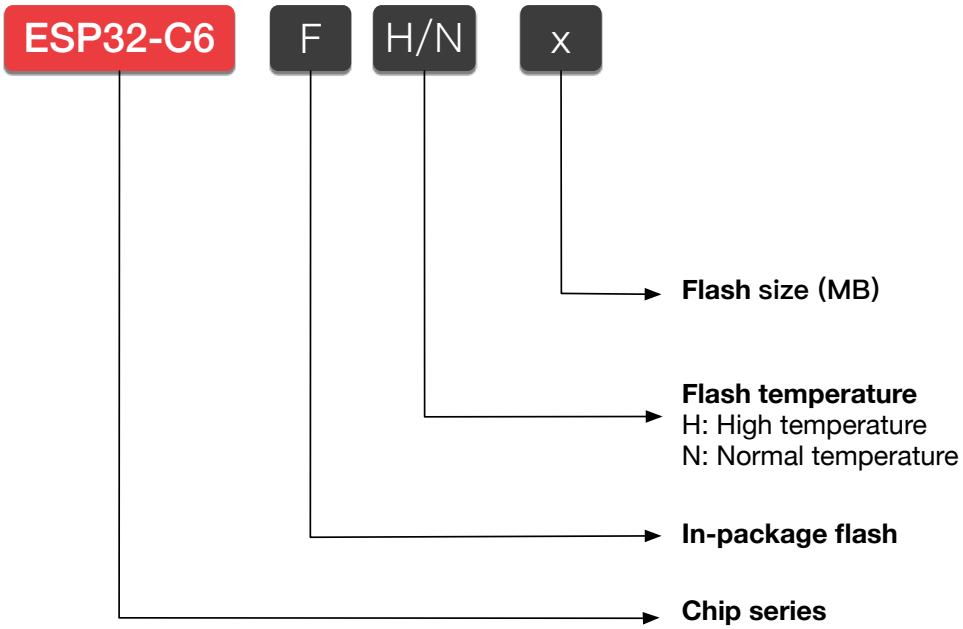



**Figure 1-1. ESP32-C6 Series Nomenclature**

## 1.2 Comparison

**Table 1-1. ESP32-C6 Series Comparison**

| Ordering Code [1] | In-Package Flash | Ambient Temp. [2] | Package |
|---|---|---|---|
| ESP32-C6 | — [3] | –40 ~ 105 °C | QFN40 (5×5 mm) |
| ESP32-C6FH4 | 4 MB (Quad SPI) [4, 5] | –40 ~ 105 °C | QFN32 (5×5 mm) |

[1] For details on chip marking and packing, see Section 7 *Packaging*.

[2] Ambient temperature specifies the recommended temperature range of the environment immediately outside an Espressif chip.

[3] Can connect a flash outside the chip package. For details, see Section 4.1.2.2 *External Memory*.

[4] For details about SPI modes, see Section 2.6 *Pin Mapping Between Chip and Flash*.

[5] For information about in-package flash, see also Section 4.1.2.1 *Internal Memory*. By default, the SPI flash on the chip operates at a maximum clock frequency of 80 MHz and does not support the auto suspend feature. If you have a requirement for a higher flash clock frequency of 120 MHz or if you need the flash auto suspend feature, please contact us.

# 2 Pins

## 2.1 Pin Layout

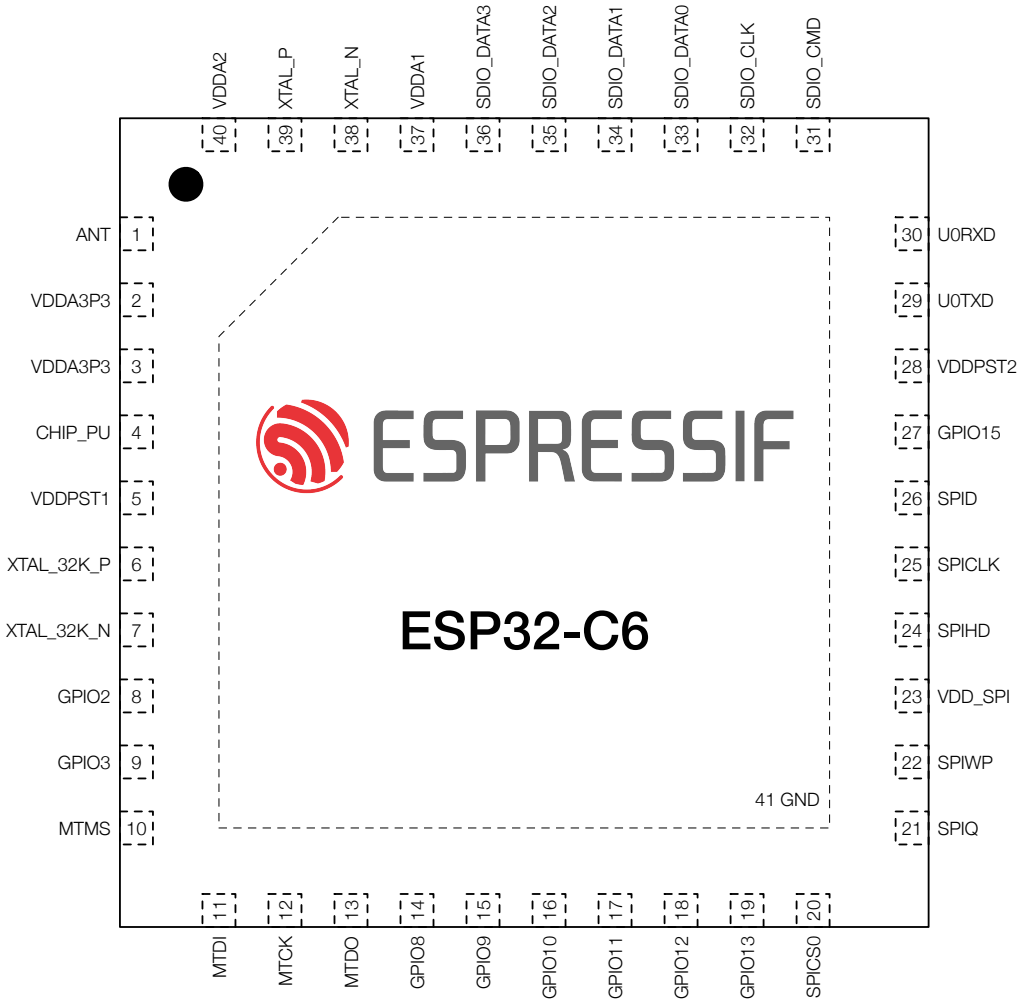



**Figure 2-1. ESP32-C6 Pin Layout (QFN40, Top View)**

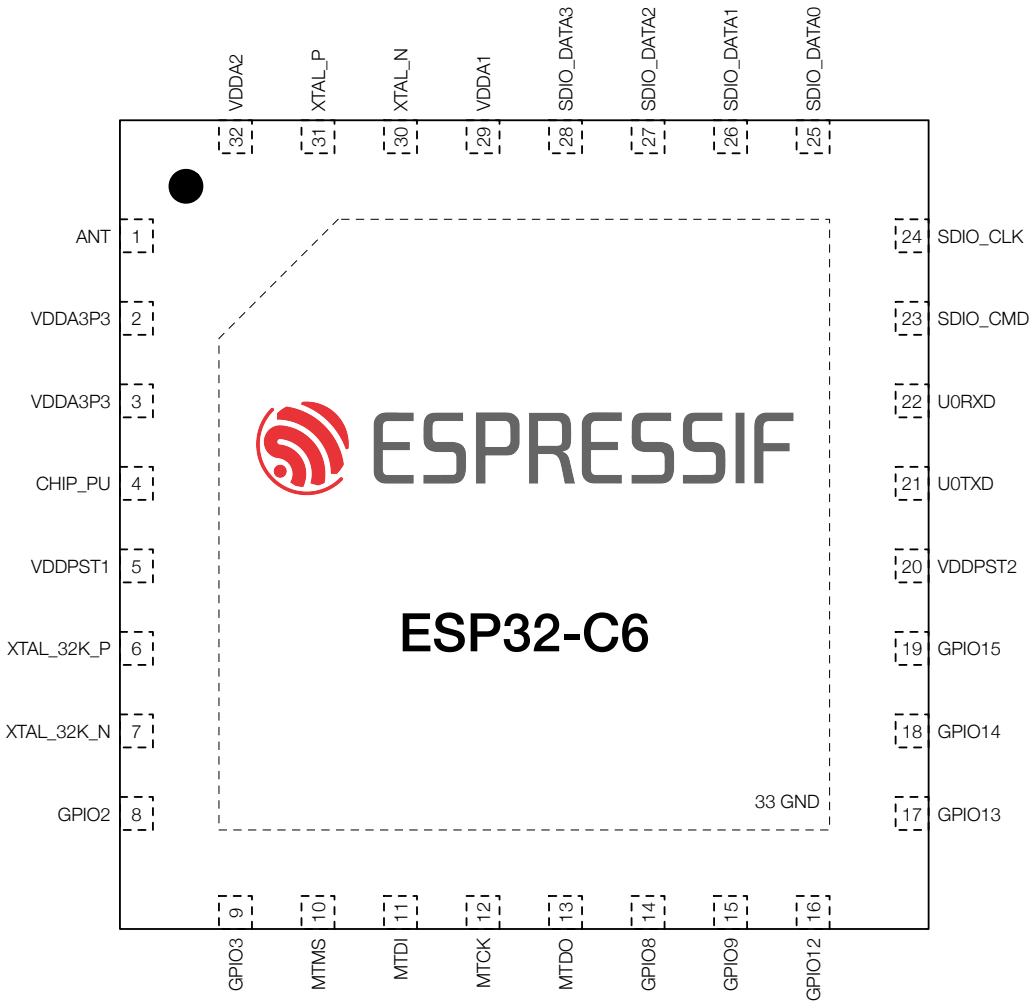



**Figure 2-2. ESP32-C6 Pin Layout (QFN32, Top View)**

## 2.2 Pin Overview

The ESP32-C6 chip integrates multiple peripherals that require communication with the outside world. To keep the chip package size reasonably small, the number of available pins has to be limited. So the only way to route all the incoming and outgoing signals is through pin multiplexing. Pin muxing is controlled via software programmable registers (see *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*).

All in all, the ESP32-C6 chip has the following types of pins:

- **IO pins** with the following predefined sets of functions to choose from:
  - **Each** IO pin has predefined **IO MUX functions** – see Table 2-4 *QFN40 IO MUX Pin Functions* or Table 2-5 *QFN32 IO MUX Pin Functions*
  - **Some** IO pins have predefined **LP IO MUX functions** – see Table 2-7 *LP IO MUX Functions*
  - **Some** IO pins have predefined **analog functions** – see Table 2-9 *Analog Functions*

  *Predefined functions* means that each IO pin has a set of direct connections to certain signals from on-chip peripherals. During run-time, the user can configure which peripheral signal from a predefined set to connect to a certain pin at a certain time via memory mapped registers (see the TRM).
- **Analog pins** that have exclusively-dedicated **analog functions** – see Table 2-10 *Analog Pins*
- **Power pins** that supply power to the chip components and non-power pins – see Table 2-11 *Power Pins*

Table 2-1 *QFN40 Pin Overview* or Table 2-2 *QFN32 Pin Overview* gives an overview of all the pins. For more information, see the respective sections for each pin type below, or Appendix A – ESP32-C6 Consolidated Pin Overview.

**Table 2-1. QFN40 Pin Overview**

| Pin No. | Pin Name | Pin Type | Pin Providing Power [2-4] | Pin Settings [5, 6] | | Pin Function Sets [1] | | |
|---|---|---|---|---|---|---|---|---|
| | | | | At Reset | After Reset | IO MUX | LP IO MUX | Analog |
| 1 | ANT | Analog | | | | | | |
| 2 | VDDA3P3 | Power | | | | | | |
| 3 | VDDA3P3 | Power | | | | | | |
| 4 | CHIP_PU | Analog | VDDPST1 | | | | | |
| 5 | VDDPST1 | Power | | | | | | |
| 6 | XTAL_32K_P | IO | VDDPST1 | | | **IO MUX** | LP IO MUX | Analog |
| 7 | XTAL_32K_N | IO | VDDPST1 | | | **IO MUX** | LP IO MUX | Analog |
| 8 | GPIO2 | IO | VDDPST1 | IE | IE | **IO MUX** | LP IO MUX | Analog |
| 9 | GPIO3 | IO | VDDPST1 | IE | IE | **IO MUX** | LP IO MUX | Analog |
| 10 | MTMS | IO | VDDPST1 | IE | IE | **IO MUX** | LP IO MUX | Analog |
| 11 | MTDI | IO | VDDPST1 | IE | IE | **IO MUX** | LP IO MUX | Analog |
| 12 | MTCK | IO | VDDPST1 | | IE, WPU [5] | **IO MUX** | LP IO MUX | Analog |
| 13 | MTDO | IO | VDDPST1 | | IE | **IO MUX** | LP IO MUX | |
| 14 | GPIO8 | IO | VDDPST2 | IE | IE | **IO MUX** | | |
| 15 | GPIO9 | IO | VDDPST2 | IE, WPU | IE, WPU | **IO MUX** | | |
| 16 | GPIO10 | IO | VDDPST2 | | IE | **IO MUX** | | |
| 17 | GPIO11 | IO | VDDPST2 | | IE | **IO MUX** | | |
| 18 | GPIO12 | IO | VDDPST2 | | IE | IO MUX | | **Analog** |

**Cont'd on next page**

Table 2-1 – cont'd from previous page

| Pin No. | Pin Name | Pin Type | Pin Providing Power [2-4] | Pin Settings [5, 6] | | Pin Function Sets [1] | | |
|---|---|---|---|---|---|---|---|---|
| | | | | At Reset | After Reset | IO MUX | LP IO MUX | Analog |
| 19 | GPIO13 | IO | VDDPST2 | USB_PU | IE, USB_PU | IO MUX | | **Analog** |
| 20 | SPICS0 | IO | VDD_SPI | WPU | IE, WPU | **IO MUX** | | |
| 21 | SPIQ | IO | VDD_SPI | WPU | IE, WPU | **IO MUX** | | |
| 22 | SPIWP | IO | VDD_SPI | WPU | IE, WPU | **IO MUX** | | |
| 23 | VDD_SPI | Power/IO | — | | | IO MUX | | **Analog** |
| 24 | SPIHD | IO | VDD_SPI | WPU | IE, WPU | **IO MUX** | | |
| 25 | SPICLK | IO | VDD_SPI | WPU | IE, WPU | **IO MUX** | | |
| 26 | SPID | IO | VDD_SPI | WPU | IE, WPU | **IO MUX** | | |
| 27 | GPIO15 | IO | VDDPST2 | IE | IE | **IO MUX** | | |
| 28 | VDDPST2 | Power | | | | | | |
| 29 | U0TXD | IO | VDDPST2 | | WPU [6] | **IO MUX** | | |
| 30 | U0RXD | IO | VDDPST2 | | IE, WPU | **IO MUX** | | |
| 31 | SDIO_CMD | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 32 | SDIO_CLK | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 33 | SDIO_DATA0 | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 34 | SDIO_DATA1 | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 35 | SDIO_DATA2 | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 36 | SDIO_DATA3 | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 37 | VDDA1 | Power | | | | | | |
| 38 | XTAL_N | Analog | | | | | | |
| 39 | XTAL_P | Analog | | | | | | |
| 40 | VDDA2 | Power | | | | | | |
| 41 | GND | Power | | | | | | |

Table 2-2. QFN32 Pin Overview

| Pin No. | Pin Name | Pin Type | Pin Providing Power [2-4] | Pin Settings [5, 6] | | Pin Function Sets [1] | | |
|---|---|---|---|---|---|---|---|---|
| | | | | At Reset | After Reset | IO MUX | LP IO MUX | Analog |
| 1 | ANT | Analog | | | | | | |
| 2 | VDDA3P3 | Power | | | | | | |
| 3 | VDDA3P3 | Power | | | | | | |
| 4 | CHIP_PU | Analog | VDDPST1 | | | | | |
| 5 | VDDPST1 | Power | | | | | | |
| 6 | XTAL_32K_P | IO | VDDPST1 | | | **IO MUX** | LP IO MUX | Analog |
| 7 | XTAL_32K_N | IO | VDDPST1 | | | **IO MUX** | LP IO MUX | Analog |
| 8 | GPIO2 | IO | VDDPST1 | IE | IE | **IO MUX** | LP IO MUX | Analog |
| 9 | GPIO3 | IO | VDDPST1 | IE | IE | **IO MUX** | LP IO MUX | Analog |
| 10 | MTMS | IO | VDDPST1 | IE | IE | **IO MUX** | LP IO MUX | Analog |
| 11 | MTDI | IO | VDDPST1 | IE | IE | **IO MUX** | LP IO MUX | Analog |
| 12 | MTCK | IO | VDDPST1 | | IE, WPU [5] | **IO MUX** | LP IO MUX | Analog |
| 13 | MTDO | IO | VDDPST1 | | IE | **IO MUX** | LP IO MUX | |
| 14 | GPIO8 | IO | VDDPST2 | IE | IE | **IO MUX** | | |
| 15 | GPIO9 | IO | VDDPST2 | IE, WPU | IE, WPU | **IO MUX** | | |
| 16 | GPIO12 | IO | VDDPST2 | | IE | IO MUX | | **Analog** |

**Cont'd on next page**

Table 2-2 – cont'd from previous page

| Pin No. | Pin Name | Pin Type | Pin Providing Power [2-4] | Pin Settings [5, 6] | | Pin Function Sets [1] | | |
|---|---|---|---|---|---|---|---|---|
| | | | | At Reset | After Reset | IO MUX | LP IO MUX | Analog |
| 17 | GPIO13 | IO | VDDPST2 | USB_PU | IE, USB_PU | IO MUX | | **Analog** |
| 18 | GPIO14 | IO | VDDPST2 | | IE | **IO MUX** | | |
| 19 | GPIO15 | IO | VDDPST2 | IE | IE | **IO MUX** | | |
| 20 | VDDPST2 | Power | | | | | | |
| 21 | U0TXD | IO | VDDPST2 | | WPU [6] | **IO MUX** | | |
| 22 | U0RXD | IO | VDDPST2 | | IE, WPU | **IO MUX** | | |
| 23 | SDIO_CMD | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 24 | SDIO_CLK | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 25 | SDIO_DATA0 | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 26 | SDIO_DATA1 | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 27 | SDIO_DATA2 | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 28 | SDIO_DATA3 | IO | VDDPST2 | WPU | IE | **IO MUX** | | |
| 29 | VDDA1 | Power | | | | | | |
| 30 | XTAL_N | Analog | | | | | | |
| 31 | XTAL_P | Analog | | | | | | |
| 32 | VDDA2 | Power | | | | | | |
| 33 | GND | Power | | | | | | |

1. **Bold** marks the pin function set in which a pin has its default function in the default boot mode. See Section 3.1 *Chip Boot Mode Control*.
2. In column **Pin Providing Power**, regarding pins powered by VDD_SPI:
   - Power actually comes from the internal power rail supplying power to VDD_SPI. For details, see Section 2.5.2 *Power Scheme*.
3. Except for GPIO12 and GPIO13 whose default drive strength is 40 mA, the default drive strength for all the other pins is 20 mA.
4. Column **Pin Settings** shows predefined settings at reset and after reset with the following abbreviations:
   - IE – input enabled
   - WPU – internal weak pull-up resistor enabled
   - WPD – internal weak pull-down resistor enabled
   - USB_PU – USB pull-up resistor enabled
     - By default, the USB function is enabled for USB pins (i.e., GPIO12 and GPIO13), and the pin pull-up is decided by the USB pull-up resistor. The USB pull-up resistor is controlled by USB_SERIAL_JTAG_DP/DM_PULLUP and the pull-up value is controlled by USB_SERIAL_JTAG_PULLUP_VALUE. For details, see *ESP32-C6 Technical Reference Manual* > Chapter *USB Serial/JTAG Controller*).
     - When the USB function is disabled, USB pins are used as regular GPIOs. At reset, GPIO13's internal weak pull-up resistor is disabled by default. After reset, GPIO13's internal weak pull-up resistor is enabled by default. A pin's internal weak pull-up and pull-down resistors are configurable by IO_MUX_FUN_WPU/WPD.
5. Depends on the value of EFUSE_DIS_PAD_JTAG
   - **0** - default value. Input enabled, and internal weak pull-up resistor enabled (IE & WPU)
   - **1** - input enabled (IE)
6. Output enabled

## 2.3 IO Pins

### 2.3.1 IO MUX Pin Functions

The IO MUX allows multiple input/output signals to be connected to a single input/output pin. Each IO pin of ESP32-C6 can be connected to one of the three signals (IO MUX functions, i.e. F0-F2), as listed in Table 2-4 *QFN40 IO MUX Pin Functions* and Table 2-5 *QFN32 IO MUX Pin Functions*.

Among the three sets of signals:

- Some are routed via the GPIO Matrix (**GPIO0, GPIO1, etc.**), which incorporates internal signal routing circuitry for mapping signals programmatically. It gives the pin access to almost any peripheral signals. However, the flexibility of programmatic mapping comes at a cost as it might affect the latency of routed signals. For details about connecting to peripheral signals via GPIO Matrix, see *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.
- Some are directly routed from certain peripherals (**U0TXD, MTCK, etc.**), including UART0/1, JTAG, SPI0/1, SPI2, and SDIO - see Table 2-3 *Peripheral Signals Routed via IO MUX*.

**Table 2-3. Peripheral Signals Routed via IO MUX**

| Pin Function | Signal | Description |
|---|---|---|
| U0TXD<br>U0RXD | Transmit data<br>Receive data | UART0 interface |
| MTCK<br>MTDO<br>MTDI<br>MTMS | Test clock<br>Test Data Out<br>Test Data In<br>Test Mode Select | JTAG interface for debugging |
| SPIQ<br>SPID<br>SPIHD<br>SPIWP<br>SPICLK<br>SPICS0 | Data out<br>Data in<br>Hold<br>Write protect<br>Clock<br>Chip select | 3.3 V SPI0/1 interface for connection to in-package or off-package flash via the SPI bus. It supports 1-, 2-, 4-line SPI modes. See also Section 2.6 *Pin Mapping Between Chip and Flash* |
| FSPIQ<br>FSPID<br>FSPIHD<br>FSPIWP<br>FSPICLK<br>FSPICS… | Data out<br>Data in<br>Hold<br>Write protect<br>Clock<br>Chip select | SPI2 interface for fast SPI connection. It supports 1-, 2-, 4-line SPI modes |
| SDIO_CMD<br>SDIO_CLK<br>SDIO_DATA… | Command<br>Clock<br>Data | SDIO 2.0 interface |

Table 2-4 *QFN40 IO MUX Pin Functions* and Table 2-5 *QFN32 IO MUX Pin Functions* show the IO MUX functions of IO pins.

Table 2-4. QFN40 IO MUX Pin Functions

| Pin No. | IO MUX / GPIO Name [2] | IO MUX Function [1, 2, 3] | | | | | |
|---|---|---|---|---|---|---|---|
| | | F0 | Type [3] | F1 | Type | F2 | Type |
| 6 | GPIO0 | **GPIO0** | I/O/T | GPIO0 | I/O/T | | |
| 7 | GPIO1 | **GPIO1** | I/O/T | GPIO1 | I/O/T | | |
| 8 | GPIO2 | **GPIO2** | I/O/T | GPIO2 | I/O/T | FSPIQ | I1/O/T |
| 9 | GPIO3 | **GPIO3** | I/O/T | GPIO3 | I/O/T | | |
| 10 | GPIO4 | **MTMS** | I1 | GPIO4 | I/O/T | FSPIHD | I1/O/T |
| 11 | GPIO5 | **MTDI** | I1 | GPIO5 | I/O/T | FSPIWP | I1/O/T |
| 12 | GPIO6 | **MTCK** | I1 | GPIO6 | I/O/T | FSPICLK | I1/O/T |
| 13 | GPIO7 | **MTDO** | O/T | GPIO7 | I/O/T | FSPID | I1/O/T |
| 14 | GPIO8 | **GPIO8** | I/O/T | GPIO8 | I/O/T | | |
| 15 | GPIO9 | **GPIO9** | I/O/T | GPIO9 | I/O/T | | |
| 16 | GPIO10 | **GPIO10** | I/O/T | GPIO10 | I/O/T | | |
| 17 | GPIO11 | **GPIO11** | I/O/T | GPIO11 | I/O/T | | |
| 18 | GPIO12 | GPIO12 | I/O/T | GPIO12 | I/O/T | | |
| 19 | GPIO13 | GPIO13 | I/O/T | GPIO13 | I/O/T | | |
| 20 | GPIO24 | **SPICS0** | O/T | GPIO24 | I/O/T | | |
| 21 | GPIO25 | **SPIQ** | I1/O/T | GPIO25 | I/O/T | | |
| 22 | GPIO26 | SPIWP | I1/O/T | **GPIO26** | I/O/T | | |
| 23 | GPIO27 | GPIO27 | I/O/T | GPIO27 | I/O/T | | |
| 24 | GPIO28 | SPIHD | I1/O/T | **GPIO28** | I/O/T | | |
| 25 | GPIO29 | **SPICLK** | O/T | GPIO29 | I/O/T | | |
| 26 | GPIO30 | **SPID** | I1/O/T | GPIO30 | I/O/T | | |
| 27 | GPIO15 | **GPIO15** | I/O/T | GPIO15 | I/O/T | | |
| 29 | GPIO16 | **U0TXD** | O | GPIO16 | I/O/T | FSPICS0 | I1/O/T |
| 30 | GPIO17 | **U0RXD** | I1 | GPIO17 | I/O/T | FSPICS1 | O/T |
| 31 | GPIO18 | **SDIO_CMD** | I1/O/T | GPIO18 | I/O/T | FSPICS2 | O/T |
| 32 | GPIO19 | **SDIO_CLK** | I1 | GPIO19 | I/O/T | FSPICS3 | O/T |
| 33 | GPIO20 | **SDIO_DATA0** | I1/O/T | GPIO20 | I/O/T | FSPICS4 | O/T |
| 34 | GPIO21 | **SDIO_DATA1** | I1/O/T | GPIO21 | I/O/T | FSPICS5 | O/T |
| 35 | GPIO22 | **SDIO_DATA2** | I1/O/T | GPIO22 | I/O/T | | |
| 36 | GPIO23 | **SDIO_DATA3** | I1/O/T | GPIO23 | I/O/T | | |

[1] **Bold** marks the default pin functions in the default boot mode. See Section 3.1 *Chip Boot Mode Control*.

[2] Regarding highlighted cells, see Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*.

[3] Each IO MUX function (F*n*, *n* = 0 ~ 2) is associated with a *type*. The description of *type* is as follows:

- I – input. O – output. T – high impedance.
- I1 – input; if the pin is assigned a function other than F*n*, the input signal of F*n* is always **1**.
- I0 – input; if the pin is assigned a function other than F*n*, the input signal of F*n* is always **0**.

Table 2-5. QFN32 IO MUX Pin Functions

| Pin No. | IO MUX / GPIO Name [2] | IO MUX Function [1, 2, 3] | | | | | |
|---|---|---|---|---|---|---|---|
| | | F0 | Type [3] | F1 | Type | F2 | Type |
| 6 | GPIO0 | **GPIO0** | I/O/T | GPIO0 | I/O/T | | |
| 7 | GPIO1 | **GPIO1** | I/O/T | GPIO1 | I/O/T | | |
| 8 | GPIO2 | **GPIO2** | I/O/T | GPIO2 | I/O/T | FSPIQ | I1/O/T |
| 9 | GPIO3 | **GPIO3** | I/O/T | GPIO3 | I/O/T | | |
| 10 | GPIO4 | **MTMS** | I1 | GPIO4 | I/O/T | FSPIHD | I1/O/T |
| 11 | GPIO5 | **MTDI** | I1 | GPIO5 | I/O/T | FSPIWP | I1/O/T |
| 12 | GPIO6 | **MTCK** | I1 | GPIO6 | I/O/T | FSPICLK | I1/O/T |
| 13 | GPIO7 | **MTDO** | O/T | GPIO7 | I/O/T | FSPID | I1/O/T |
| 14 | GPIO8 | **GPIO8** | I/O/T | GPIO8 | I/O/T | | |
| 15 | GPIO9 | **GPIO9** | I/O/T | GPIO9 | I/O/T | | |
| 16 | GPIO12 | GPIO12 | I/O/T | GPIO12 | I/O/T | | |
| 17 | GPIO13 | GPIO13 | I/O/T | GPIO13 | I/O/T | | |
| 18 | GPIO14 | **GPIO14** | I/O/T | GPIO14 | I/O/T | | |
| 19 | GPIO15 | **GPIO15** | I/O/T | GPIO15 | I/O/T | | |
| 21 | GPIO16 | **U0TXD** | O | GPIO16 | I/O/T | FSPICS0 | I1/O/T |
| 22 | GPIO17 | **U0RXD** | I1 | GPIO17 | I/O/T | FSPICS1 | O/T |
| 23 | GPIO18 | **SDIO_CMD** | I1/O/T | GPIO18 | I/O/T | FSPICS2 | O/T |
| 24 | GPIO19 | **SDIO_CLK** | I1 | GPIO19 | I/O/T | FSPICS3 | O/T |
| 25 | GPIO20 | **SDIO_DATA0** | I1/O/T | GPIO20 | I/O/T | FSPICS4 | O/T |
| 26 | GPIO21 | **SDIO_DATA1** | I1/O/T | GPIO21 | I/O/T | FSPICS5 | O/T |
| 27 | GPIO22 | **SDIO_DATA2** | I1/O/T | GPIO22 | I/O/T | | |
| 28 | GPIO23 | **SDIO_DATA3** | I1/O/T | GPIO23 | I/O/T | | |

[1] **Bold** marks the default pin functions in the default boot mode. See Section 3.1 *Chip Boot Mode Control*.

[2] Regarding highlighted cells, see Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*.

[3] Each IO MUX function (F*n*, *n* = 0 ~ 2) is associated with a *type*. The description of *type* is as follows:

- I – input. O – output. T – high impedance.
- I1 – input; if the pin is assigned a function other than F*n*, the input signal of F*n* is always **1**.
- I0 – input; if the pin is assigned a function other than F*n*, the input signal of F*n* is always **0**.

### 2.3.2 LP IO MUX Functions

When the chip is in Deep-sleep mode, the IO MUX described in Section 2.3.1 *IO MUX Pin Functions* will not work. That is where the LP IO MUX comes in. It allows multiple input/output signals to be a single input/output pin in Deep-sleep mode, as the pin is connected to the LP system and powered by VDDPST1.

LP IO pins can be assigned to **LP functions**. They can

- Either work as LP GPIOs (**LP_GPIO0, LP_GPIO1, etc.**), connected to the LP CPU
- Or connect to LP peripheral signals (**LP_I2C_SDA, LP_I2C_SCL, etc.**) - see Table 2-6 *LP Peripheral Signals Routed via LP IO MUX*

**Table 2-6. LP Peripheral Signals Routed via LP IO MUX**

| Pin Function | Signal | Description |
|---|---|---|
| LP_I2C_SDA | Serial data | LP I2C interface |
| LP_I2C_SCL | Serial clock | |
| LP_UART_RXD | Receive | LP UART interface |
| LP_UART_TXD | Transmit | |
| LP_UART_RTSN | Request to send | |
| LP_UART_CTSN | Clear to send | |
| LP_UART_DTRN | Data set ready | |
| LP_UART_DSRN | Data terminal ready | |

Table 2-7 LP IO MUX Functions shows the LP functions of LP IO pins.

**Table 2-7. LP IO MUX Functions**

| Pin No. | LP IO Name [1, 2, 3] | LP IO MUX Function | |
|---|---|---|---|
| | | F0 | F1 |
| 6 | LP_GPIO0 | LP_GPIO0 | LP_UART_DTRN |
| 7 | LP_GPIO1 | LP_GPIO1 | LP_UART_DSRN |
| 8 | LP_GPIO2 | LP_GPIO2 | LP_UART_RTSN |
| 9 | LP_GPIO3 | LP_GPIO3 | LP_UART_CTSN |
| 10 | LP_GPIO4 | LP_GPIO4 | LP_UART_RXD |
| 11 | LP_GPIO5 | LP_GPIO5 | LP_UART_TXD |
| 12 | LP_GPIO6 | LP_GPIO6 | LP_I2C_SDA |
| 13 | LP_GPIO7 | LP_GPIO7 | LP_I2C_SCL |

[1] **Bold** marks the default pin functions in the default boot mode. See Section 3.1 *Chip Boot Mode Control*.

[2] This column lists the LP GPIO names, since LP functions are configured with LP GPIO registers that use LP GPIO numbering.

[3] Regarding highlighted cells, see Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*.

### 2.3.3 Analog Functions

Some IO pins also have **analog functions**, for analog peripherals (such as ADC) in any power mode. Internal analog signals are routed to these analog functions, see Table 2-8 *Analog Signals Routed to Analog Functions*.

**Table 2-8. Analog Signals Routed to Analog Functions**

| Pin Function | Signal | Description |
|---|---|---|
| ADC1_CH… | ADC1 channel … signal | ADC1 interface |
| XTAL_32K_N<br>XTAL_32K_P | Negative clock signal<br>Positive clock signal | 32 kHz external clock input/output connected to ESP32-C6's oscillator |
| USB_D-<br>USB_D+ | Data -<br>Data + | USB Serial/JTAG function |

Table 2-9 Analog Functions shows the analog functions of IO pins.

**Table 2-9. Analog Functions**

| QFN40 Pin No. | QFN32 Pin No. | Analog IO Name [1, 2] | Analog Function [2] F0 | Analog Function [2] F1 |
|---|---|---|---|---|
| 6 | 6 | GPIO0 | XTAL_32K_P | ADC1_CH0 |
| 7 | 7 | GPIO1 | XTAL_32K_N | ADC1_CH1 |
| 8 | 8 | GPIO2 | | ADC1_CH2 |
| 9 | 9 | GPIO3 | | ADC1_CH3 |
| 10 | 10 | GPIO4 | | ADC1_CH4 |
| 11 | 11 | GPIO5 | | ADC1_CH5 |
| 12 | 12 | GPIO6 | | ADC1_CH6 |
| 18 | 16 | GPIO12 | **USB_D-** | |
| 19 | 17 | GPIO13 | **USB_D+** | |
| 23 | — | GPIO27 | **VDD_SPI** | |

[1] **Bold** marks the default pin functions in the default boot mode. See Section 3.1 *Chip Boot Mode Control*.

[2] Regarding highlighted cells, see Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*.

### 2.3.4 Restrictions for GPIOs and LP GPIOs

All IO pins of ESP32-C6 have GPIO and some have LP GPIO pin functions. However, the IO pins are multiplexed and can be configured for different purposes based on the requirements. Some IOs have restrictions for usage. It is essential to consider the multiplexed nature and the limitations when using these IO pins.

In tables of this chapter, some pin functions are highlighted. The non-highlighted GPIO or LP GPIO pins are recommended for use first. If more pins are needed, the highlighted GPIOs or LP GPIOs should be chosen carefully to avoid conflicts with important pin functions.

The highlighted IO pins have the following important pin functions:

- GPIO – allocated for communication with flash and NOT recommended for other uses. For details, see Section 2.6 *Pin Mapping Between Chip and Flash*.
- GPIO – have one of the following important functions:
  - **Strapping pins** – need to be at certain logic levels at startup. See Section 3 *Boot Configurations*.
  - **USB_D+/-** – by default, connected to the USB Serial/JTAG Controller. To function as GPIOs, these pins need to be reconfigured.
  - **JTAG interface** – often used for debugging. See Table 2-4 *QFN40 IO MUX Pin Functions* or Table 2-5 *QFN32 IO MUX Pin Functions*. To free these pins up, the pin functions USB_D+/- of the USB Serial/JTAG Controller can be used instead. See also Section 3.4 *JTAG Signal Source Control*.
  - **UART interface** – often used for debugging. See Table 2-4 *QFN40 IO MUX Pin Functions* or Table 2-5 *QFN32 IO MUX Pin Functions*.

See also Appendix A – ESP32-C6 Consolidated Pin Overview.

## 2.4 Analog Pins

**Table 2-10. Analog Pins**

| QFN40 Pin No. | QFN32 Pin No. | Pin Name | Pin Type | Pin Function |
|---|---|---|---|---|
| 1 | 1 | ANT | I/O | RF input and output |
| 4 | 4 | CHIP_PU | — | High: on, enables the chip (powered up).<br>Low: off, disables the chip (powered down).<br>Note: Do not leave the CHIP_PU pin floating. |
| 38 | 30 | XTAL_N | — | External clock input/output connected to chip's crystal or oscillator. P/N means differential clock positive/negative. |
| 39 | 31 | XTAL_P | — | |

## 2.5 Power Supply

### 2.5.1 Power Pins

The chip is powered via the power pins described in Table 2-11 *Power Pins*.

**Table 2-11. Power Pins**

| QFN40 Pin No. | QFN32 Pin No. | Pin Name | Direction | Power Supply [1,2] Power Domain / Other | IO Pins [4] |
|---|---|---|---|---|---|
| 2 | 2 | VDDA3P3 | Input | Analog power domain | |
| 3 | 3 | VDDA3P3 | Input | Analog power domain | |
| 5 | 5 | VDDPST1 | Input | LP digital and part of analog pin power domains | LP IO |
| 23 | — | VDD_SPI [3] | Input | In-package flash (backup power line) | |
| | | | Output | In-package flash and off-package flash | |
| 28 | 20 | VDDPST2 | Input | HP digital power domain | HP IO |
| 37 | 29 | VDDA1 | Input | Analog power domain | |
| 40 | 32 | VDDA2 | Input | Analog power domain | |
| 41 | 33 | GND | — | External ground connection | |

[1] See in conjunction with Section 2.5.2 *Power Scheme*.

[2] For recommended and maximum voltage and current, see Section 5.1 *Absolute Maximum Ratings* and Section 5.2 *Recommended Operating Conditions*.

[3] To configure VDD_SPI as input or output, see *ESP32-C6 Technical Reference Manual* > Chapter *Low-power Management*.

[4] LP IO pins are those powered by VDDPST1 and so on, as shown in Figure 2-3 *ESP32-C6 Power Scheme*. See also Table 2-1 *QFN40 Pin Overview* or Table 2-2 *QFN32 Pin Overview* > Column *Pin Providing Power*.

### 2.5.2 Power Scheme

The power scheme is shown in Figure 2-3 *ESP32-C6 Power Scheme*.

The components on the chip are powered via voltage regulators.

**Table 2-12. Voltage Regulators**

| Voltage Regulator | Output | Power Supply |
|---|---|---|
| HP | 1.1 V | HP power domain |
| LP | 1.1 V | LP power domain |

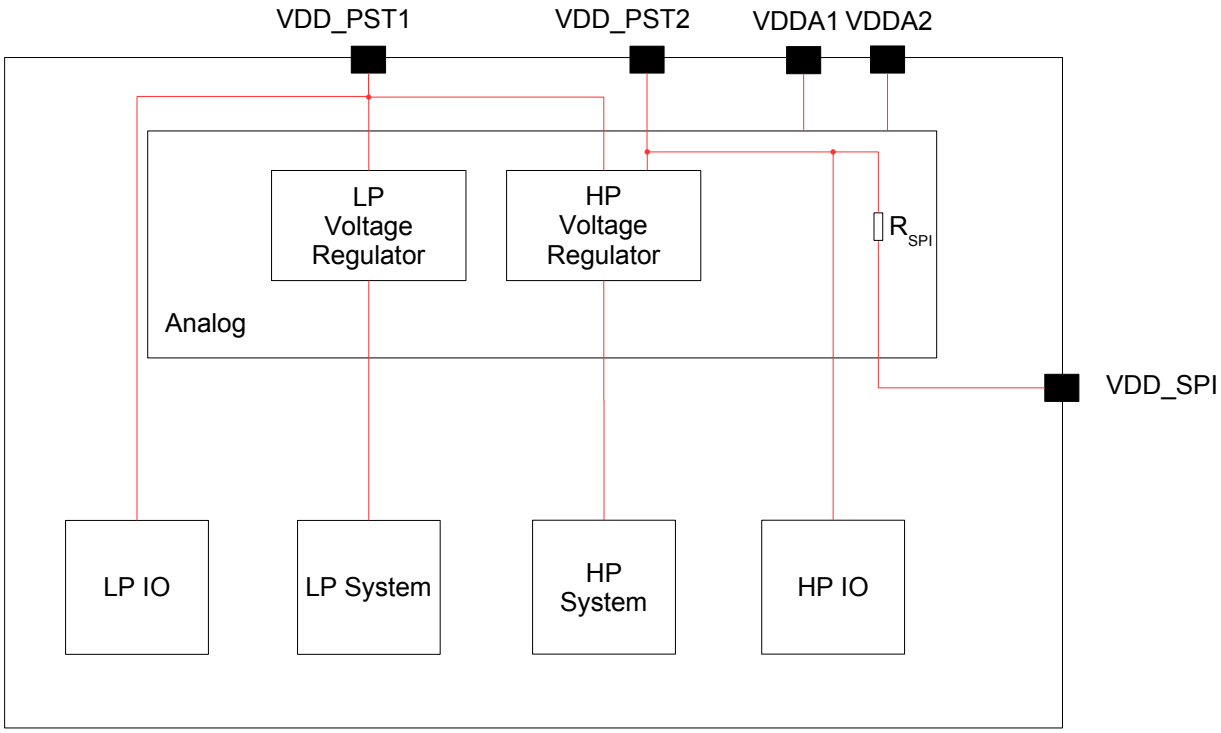



**Figure 2-3. ESP32-C6 Power Scheme**

### 2.5.3 Chip Power-up and Reset

Once the power is supplied to the chip, its power rails need a short time to stabilize. After that, CHIP_PU – the pin used for power-up and reset – is pulled high to activate the chip. For information on CHIP_PU as well as power-up and reset timing, see Figure 2-4 and Table 2-13.

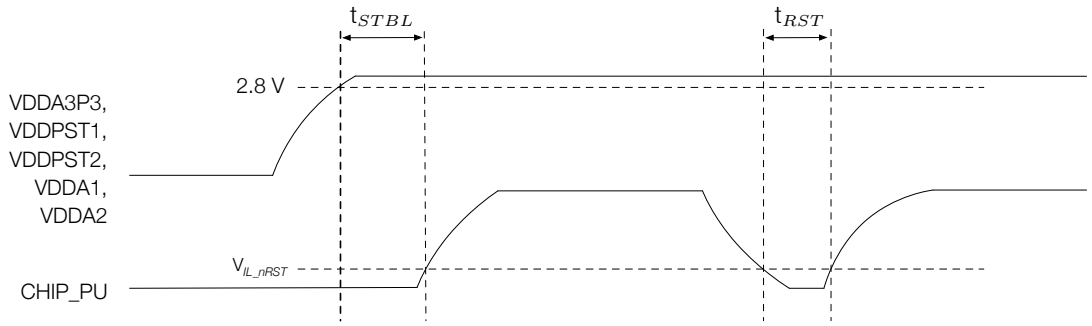



**Figure 2-4. Visualization of Timing Parameters for Power-up and Reset**

**Table 2-13. Description of Timing Parameters for Power-up and Reset**

| Parameter | Description | Min ($\mu$s) |
|---|---|---|
| $t_{STBL}$ | Time reserved for the power rails of VDDA3P3, VDDPST1, VDDPST2, VDDA1 and VDDA2 to stabilize before the CHIP_PU pin is pulled high to activate the chip | 50 |
| $t_{RST}$ | Time reserved for CHIP_PU to stay below $V_{IL\_nRST}$ to reset the chip (see Table 5-4) | 50 |

## 2.6 Pin Mapping Between Chip and Flash

Table 2-14 lists the pin mapping between the chip and off-package flash for all SPI modes.

For chip variants with in-package flash (namely variants in QFN32 package, see Table 1-1 *ESP32-C6 Series Comparison*), the pins allocated for communication with in-package flash are not routed out, but you can take Table 2-14 as a reference.

For more information on SPI controllers, see also Section 4.2.1.2 *SPI Controller*.

**Notice:**

It is not recommended to use the pins connected to flash for any other purposes.

**Table 2-14. Pin Mapping Between QFN40 Chip and Off-package Flash**

| QFN40 Pin No. | Pin Name | Single SPI Flash | Dual SPI Flash | Quad SPI / QPI Flash |
|---|---|---|---|---|
| 25 | SPICLK | CLK | CLK | CLK |
| 20 | SPICS0 | CS# | CS# | CS# |
| 26 | SPID | MOSI | SIO0 | SIO0 |
| 21 | SPIQ | MISO | SIO1 | SIO1 |
| 22 | SPIWP | WP# | | SIO2 |
| 24 | SPIHD | HOLD# | | SIO3 |

[1] SIO: Serial Data Input and Output

# 3 Boot Configurations

The chip allows for configuring the following boot parameters through strapping pins and eFuse parameters at power-up or a hardware reset, without microcontroller interaction.

- **Chip boot mode**
  - Strapping pin: GPIO8 and GPIO9
- **SDIO Sampling and Driving Clock Edge**
  - Strapping pin: MTMS and MTDI
- **ROM message printing**
  - Strapping pin: GPIO8
  - eFuse parameter: EFUSE_UART_PRINT_CONTROL and EFUSE_DIS_USB_SERIAL_JTAG_ROM_PRINT
- **JTAG signal source**
  - Strapping pin: GPIO15
  - eFuse parameter: EFUSE_DIS_PAD_JTAG, EFUSE_DIS_USB_JTAG, and EFUSE_JTAG_SEL_ENABLE

The default values of all the above eFuse parameters are 0, which means that they are not burnt. Given that eFuse is one-time programmable, once programmed to 1, it can never be reverted to 0. For how to program eFuse parameters, please refer to *ESP32-C6 Technical Reference Manual* > Chapter *eFuse Controller*.

The default values of the strapping pins, namely the logic levels, are determined by pins' internal weak pull-up/pull-down resistors at reset if the pins are not connected to any circuit, or connected to an external high-impedance circuit.

**Table 3-1. Default Configuration of Strapping Pins**

| Strapping Pin | Default Configuration | Bit Value |
|---|---|---|
| **MTMS** | Floating | – |
| **MTDI** | Floating | – |
| **GPIO8** | Floating | – |
| **GPIO9** | Weak pull-up | 1 |
| **GPIO15** | Floating | – |

To change the bit values, the strapping pins should be connected to external pull-down/pull-up resistances. If the ESP32-C6 is used as a device by a host MCU, the strapping pin voltage levels can also be controlled by the host MCU.

All strapping pins have latches. At system reset, the latches sample the bit values of their respective strapping pins and store them until the chip is powered down or shut down. The states of latches cannot be changed in any other way. It makes the strapping pin values available during the entire chip operation, and the pins are freed up to be used as regular IO pins after reset.

The timing of signals connected to the strapping pins should adhere to the *setup time* and *hold time* specifications in Table 3-2 and Figure 3-1.

**Table 3-2. Description of Timing Parameters for the Strapping Pins**

| Parameter | Description | Min (ms) |
|---|---|---|
| $t_{SU}$ | *Setup time* is the time reserved for the power rails to stabilize before the CHIP_PU pin is pulled high to activate the chip. | 0 |
| $t_{H}$ | *Hold time* is the time reserved for the chip to read the strapping pin values after CHIP_PU is already high and before these pins start operating as regular IO pins. | 3 |

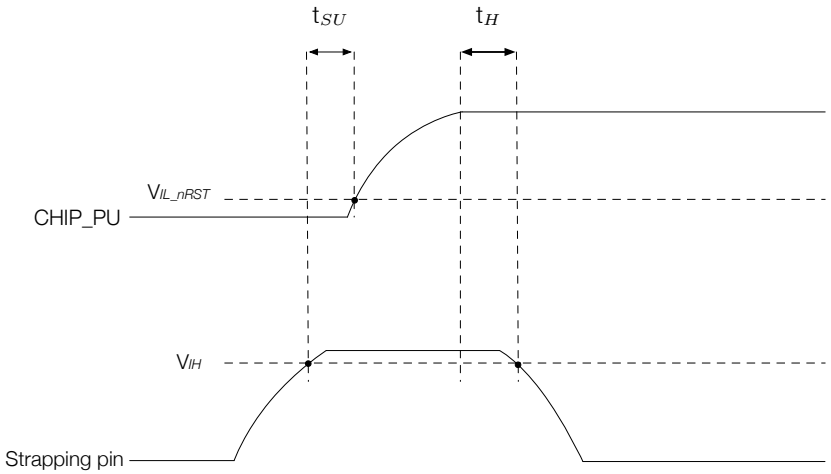



**Figure 3-1. Visualization of Timing Parameters for the Strapping Pins**

## 3.1 Chip Boot Mode Control

GPIO8 and GPIO9 control the boot mode after the reset is released. See Table 3-3 *Chip Boot Mode Control*.

**Table 3-3. Chip Boot Mode Control**

| Boot Mode | GPIO8 | GPIO9 |
|---|---|---|
| **SPI boot mode** | Any value | 1 |
| Joint download boot mode [2] | 1 | 0 |

[1] **Bold** marks the default value and configuration.

[2] Joint Download Boot mode supports the following download methods:

- USB-Serial-JTAG Download Boot
- UART Download Boot
- SDIO Download Boot

## 3.2 SDIO Sampling and Driving Clock Edge Control

The strapping pin MTMS and MTDI can be used to decide on which clock edge to sample signals and drive output lines. See Table 3-4 *SDIO Input Sampling Edge/Output Driving Edge Control*.

**Table 3-4. SDIO Input Sampling Edge/Output Driving Edge Control**

| Edge behavior | MTMS | MTDI |
|---|---|---|
| Falling edge sampling, falling edge output | 0 | 0 |
| Falling edge sampling, rising edge output | 0 | 1 |
| Rising edge sampling, falling edge output | 1 | 0 |
| Rising edge sampling, rising edge output | 1 | 1 |

[1] MTMS and MTDI are floating by default, so above are not default configurations.

## 3.3 ROM Messages Printing Control

During the boot process, the messages by the ROM code can be printed to:

- **(Default) UART0 and USB Serial/JTAG controller**
- USB Serial/JTAG controller
- UART0

EFUSE_UART_PRINT_CONTROL and GPIO8 control ROM messages printing to **UART0** as shown in Table 3-5 *UART0 ROM Message Printing Control*.

**Table 3-5. UART0 ROM Message Printing Control**

| UART0 ROM Code Printing | EFUSE_UART_PRINT_CONTROL | GPIO8 |
|---|---|---|
| **Enabled** | **0** | Ignored |
| | 1 | 0 |
| | 2 | 1 |
| Disabled | 1 | 1 |
| | 2 | 0 |
| | 3 | Ignored |

[1] **Bold** marks the default value and configuration.

EFUSE_DIS_USB_SERIAL_JTAG_ROM_PRINT controls the printing to **USB Serial/JTAG controller** as shown in Table 3-6 *USB Serial/JTAG ROM Message Printing Control*.

Table 3-6. USB Serial/JTAG ROM Message Printing Control

| USB Serial/JTAG ROM Code Printing | EFUSE_DIS_USB_SERIAL_JTAG [2] | EFUSE_DIS_USB_SERIAL_JTAG_ROM_PRINT |
|---|---|---|
| **Enabled** | **0** | **0** |
| Disabled | 0 | 1 |
| | 1 | Ignored |

[1] **Bold** marks the default value and configuration.

[2] EFUSE_DIS_USB_SERIAL_JTAG controls whether to disable USB Serial/JTAG.

## 3.4 JTAG Signal Source Control

The strapping pin GPIO15 can be used to control the source of JTAG signals during the early boot process. This pin does not have any internal pull resistors and the strapping value must be controlled by the external circuit that cannot be in a high impedance state.

As Table 3-7 *JTAG Signal Source Control* shows, GPIO15 is used in combination with EFUSE_DIS_PAD_JTAG, EFUSE_DIS_USB_JTAG and EFUSE_JTAG_SEL_ENABLE.

Table 3-7. JTAG Signal Source Control

| JTAG Signal Source | EFUSE_DIS_PAD_JTAG | EFUSE_DIS_USB_JTAG | EFUSE_JTAG_SEL_ENABLE | GPIO15 |
|---|---|---|---|---|
| **USB Serial/JTAG Controller** | **0** | **0** | **0** | Ignored |
| | 0 | 0 | 1 | 1 |
| | 1 | 0 | Ignored | Ignored |
| JTAG pins [2] | 0 | 0 | 1 | 0 |
| | 0 | 1 | Ignored | Ignored |
| JTAG is disabled | 1 | 1 | Ignored | Ignored |

[1] **Bold** marks the default value and configuration.

[2] JTAG pins refer to MTDI, MTCK, MTMS, and MTDO.

# 4 Functional Description

## 4.1 System

This section describes the core of the chip's operation, covering its microprocessor, memory organization, system components, and security features.

### 4.1.1 Microprocessor and Master

This subsection describes the core processing units within the chip and their capabilities.

#### 4.1.1.1 High-Performance CPU

The ESP-RISC-V CPU (HP CPU) is a high-performance 32-bit core based on the RISC-V instruction set architecture (ISA) comprising base integer (I), multiplication/division (M), atomic (A) and compressed (C) standard extensions.

**Feature List**

- Four-stage pipeline that supports an operating clock frequency up to 160 MHz
- RV32IMAC ISA (instruction set architecture)
- Compatible with RISC-V ISA Manual Volume I: Unprivileged ISA Version 2.2 and RISC-V ISA Manual, Volume II: Privileged Architecture, Version 1.10
- Zero wait cycle access to on-chip SRAM and Cache for program and data access over IRAM/DRAM interface
- Branch target buffer (BTB) with static branch prediction
- User (U) mode support along with interrupt delegation
- Interrupt controller with up to 28 external vectored interrupts for both M and U modes with 16 programmable priority and threshold levels
- Core local interrupts (CLINT) dedicated for each privilege mode
- Debug module (DM) compliant with the specification RISC-V External Debug Support Version 0.13 with external debugger support over an industry-standard JTAG/USB port
- Support for instruction trace, see Section 4.1.1.2 *RISC-V Trace Encoder*
- Hardware trigger compliant to the specification RISC-V External Debug Support Version 0.13 with up to 4 breakpoints/watchpoints
- Physical memory protection (PMP) and attributes (PMA) for up to 16 configurable regions

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *High-Performance CPU*.

#### 4.1.1.2 RISC-V Trace Encoder

The RISC-V Trace Encoder in the ESP32-C6 chip provides a way to capture detailed trace information from the High-Performance CPU's execution, enabling deeper analysis and optimization of the system. It connects to the

HP CPU's instruction trace interface and compresses the information into smaller packets, which are then stored in internal SRAM.

**Feature List**

- Compatible with RISC-V Processor Trace Version 1.0
- Synchronization packets sent every few clock cycles or packets
- Zero bytes as anchor tags to identify boundaries between data packets
- Configurable memory writing mode: loop mode or non-loop mode
- Trace lost status to indicate packet loss
- Automatic restart after packet loss

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *RISC-V Trace Encoder (TRACE)*.

#### 4.1.1.3 Low-Power CPU

The ESP32-C6 Low-Power CPU (LP CPU) is a 32-bit processor based on the RISC-V ISA comprising integer (I), multiplication/division (M), atomic (A), and compressed (C) standard extensions. It is designed for ultra-low power consumption and is capable of staying powered on during Deep-sleep mode when the HP CPU is powered down.

**Feature List**

- Two-stage pipeline that supports a clock frequency of up to 20 MHz
- RV32IMAC ISA (instruction set architecture)
- 19 vector interrupts
- Debug module compliant with RISC-V External Debug Support Version 0.13 with external debugger support over an industry-standard JTAG/USB port
- Hardware trigger compliant with RISC-V External Debug Support Version 0.13 with up to 2 breakpoints/watchpoints
- 32-bit AHB system bus for peripheral and memory access
- Core performance metric events
- Able to wake up the HP CPU and send an interrupt to it
- Access to HP memory and LP memory
- Access to the entire peripheral address space

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Low-Power CPU*.

#### 4.1.1.4 GDMA Controller

The GDMA Controller is a General Direct Memory Access (GDMA) controller that allows peripheral-to-memory, memory-to-peripheral, and memory-to-memory data transfer with the CPU's intervention. The GDMA has six

independent channels, three transmit and three receive. These channels are shared by peripherals with the GDMA feature, such as SPI2, UHCI (UART0/UART1), I2S, AES, SHA, ADC, and PARLIO.

**Feature List**

- Programmable length of data to be transferred in bytes
- Linked list of descriptors for efficient data transfer management
- INCR burst transfer when accessing internal RAM for improved performance
- Access to an address space of up to 384 KB in internal RAM
- Software-configurable selection of peripheral requesting service
- Fixed-priority and round-robin channel arbitration schemes for managing bandwidth
- Support for Event Task Matrix

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *GDMA Controller (DMA)*.

### 4.1.2 Memory Organization

This subsection describes the memory arrangement to explain how data is stored, accessed, and managed for efficient operation.

Figure 4-1 illustrates the address mapping structure of ESP32-C6.

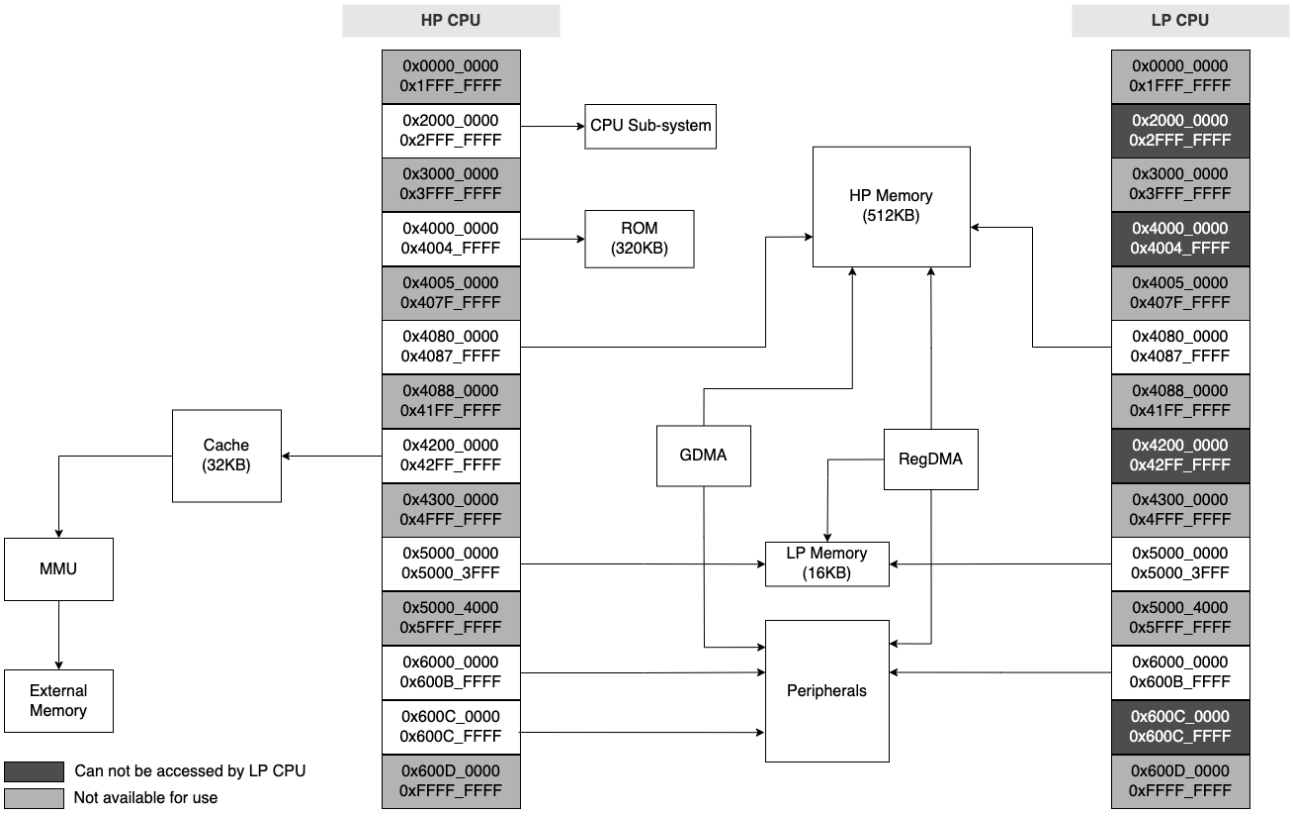



**Figure 4-1. Address Mapping Structure**

#### 4.1.2.1 Internal Memory

The internal memory of ESP32-C6 refers to the memory integrated on the chip die or in the chip package, including ROM, SRAM, eFuse, and flash.

**Feature List**

- 320 KB of ROM for booting and core functions
- 512 KB of high-performance SRAM (HP SRAM) for data and instructions
- 16 KB of low-power SRAM (LP SRAM) that can be accessed by HP CPU or LP CPU. It can retain data in Deep-sleep mode
- 4096-bit eFuse memory, with 1792 bits available for users. See also Section 4.1.2.3 *eFuse Controller*
- In-package flash
  - See flash size in Chapter 1 *ESP32-C6 Series Comparison*
  - More than 100,000 program/erase cycles
  - More than 20 years of data retention time
  - Clock frequency up to 80 MHz by default

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *System and Memory*.

#### 4.1.2.2 External Memory

ESP32-C6 allows connection to memories outside the chip's package via the SPI, Dual SPI, Quad SPI, and QPI interfaces.

**Feature List**

- Support connection to off-package flash of 16 MB at most
  - Support hardware encryption/decryption based on XTS-AES
  - Up to 16 MB of CPU instruction memory space can map into flash as individual blocks of 64 KB. 32-bit fetch is supported
  - Up to 16 MB of CPU data memory space can map into flash as individual blocks of 64 KB. 8-bit, 16-bit and 32-bit reads are supported
- External memory accessed via a 32 KB read-only cache
  - Four-way set associative
  - 32-byte cache block
  - Critical word first and early restart

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *System and Memory*.

#### 4.1.2.3 eFuse Controller

The eFuse memory is a one-time programmable memory that stores parameters and user data, and the eFuse controller of ESP32-C6 is used to program and read this eFuse memory.

**Feature List**

- Configure write protection for some blocks
- Configure read protection for some blocks
- Various hardware encoding schemes against data corruption

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *eFuse Controller*.

### 4.1.3 System Components

This subsection describes the essential components that contribute to the overall functionality and control of the system.

#### 4.1.3.1 IO MUX and GPIO Matrix

The IO MUX and GPIO Matrix in the ESP32-C6 chip provide flexible routing of peripheral input and output signals to the GPIO pins. These peripherals enhance the functionality and performance of the chip by allowing the configuration of I/O, support for multiplexing, and signal synchronization for peripheral inputs.

**Feature List**

- 30 or 22 GPIO pins for general-purpose I/O or connection to internal peripheral signals
- GPIO matrix:
  - Routing 85 peripheral input and 93 output signals to any GPIO pin
  - Signal synchronization for peripheral inputs based on IO MUX operating clock
  - GPIO Filter hardware for input signal filtering
  - Glitch Filter hardware for second time filtering on input signal
  - Sigma delta modulated (SDM) output
- IO MUX for directly connecting certain digital signals (SPI, JTAG, UART) to pins
- LP IO MUX for controlling eight LP GPIO pins (GPIO0 ~ GPIO7) used by peripherals in the LP system
- Support for Event Task Matrix

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.1.3.2 Reset

The ESP32-C6 chip provides four types of reset that occur at different levels, namely CPU Reset, Core Reset, System Reset, and Chip Reset. Except for Chip Reset, all reset types preserve the data stored in internal memory.

**Feature List**

- Four types of reset:
  - CPU Reset – Resets the CPU core
  - Core Reset – Resets the whole digital system except for the LP system
  - System reset – Resets the whole digital system, including the LP system
  - Chip reset – Resets the whole chip
- Reset trigger:
  - Directly by hardware
  - Via software by configuring the corresponding registers of the CPU
- Support for retrieving reset cause

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Reset and Clock*.

#### 4.1.3.3 Clock

The ESP32-C6 chip has clocks sourced from oscillators, RC circuits, and PLL circuits, which are then processed by dividers or selectors. The clocks can be classified into high speed clocks for devices working at higher frequencies and slow speed clocks for low-power systems and some peripherals.

**Feature List**

- High speed clocks for HP system
  - 40 MHz external crystal clock

    **Note:**
    The chip cannot operate without the external crystal clock.

  - 480 MHz internal PLL clock
- Slow speed clocks for LP system and some peripherals working in low-power mode
  - 32 kHz external crystal clock
  - Internal fast RC oscillator with adjustable frequency (17.5 MHz by default)
  - Internal slow RC oscillator with adjustable frequency (136 kHz by default)
  - 32 kHz internal slow RC oscillator
  - External slow clock input through XTAL_32K_P (32 kHz by default)

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Reset and Clock*.

#### 4.1.3.4 Interrupt Matrix

The Interrupt Matrix in the ESP32-C6 chip routes interrupt requests generated by various peripherals to CPU interrupts.

**Feature List**

- 77 peripheral interrupt sources accepted as input
- 31 CPU peripheral interrupts generated to CPU as output
- Current interrupt status query of peripheral interrupt sources
- Multiple interrupt sources mapping to a single CPU interrupt (i.e., shared interrupts)

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Interrupt Matrix*.

#### 4.1.3.5 Event Task Matrix

The Event Task Matrix (ETM) allows events from any specified peripheral to be mapped to tasks of any specified peripheral, enabling peripherals to execute specified tasks without CPU intervention. Peripherals supporting ETM include GPIO, LED PWM, general-purpose timers, RTC Timer, system timer, MCPWM, temperature sensor, ADC, I2S, LP CPU, GDMA, and PMU.

**Feature List**

- 50 channels that can be enabled and configured independently
- Receive 124 events from multiple peripherals
- Generate 130 tasks for multiple peripherals

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Event Task Matrix*.

#### 4.1.3.6 System Timer

The System Timer (SYSTIMER) in the ESP32-C6 chip is a 52-bit timer that can be used to generate tick interrupts for the operating system or as a general timer to generate periodic or one-time interrupts.

**Feature List**

- Two 52-bit counters and three 52-bit comparators
- 52-bit alarm values and 26-bit alarm periods
- Two modes to generate alarms: target mode and period mode
- Three comparators generating three independent interrupts based on configured alarm value or alarm period
- Ability to load back sleep time recorded by RTC timer via software after Deep-sleep or Light-sleep
- Counters can be stalled if the CPU is stalled or in OCD mode
- Real-time alarm events

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *System Timer*.

#### 4.1.3.7 Power Management Unit

The ESP32-C6 has an advanced Power Management Unit (PMU). It can be flexibly configured to power up different power domains of the chip to achieve the best balance between chip performance, power consumption, and wakeup latency.

The integrated LP CPU allow the ESP32-C6 to operate in Deep-sleep mode with most of the power domains turned off, thus achieving extremely low-power consumption.

Configuring the PMU is a complex procedure. To simplify power management for typical scenarios, there are the following **predefined power modes** that power up different combinations of power domains:

- **Active mode** – The HP CPU, RF circuits, and all peripherals are on. The chip can process data, receive, transmit, and listen.
- **Modem-sleep mode** – The HP CPU is on, but the clock frequency can be reduced. The wireless connections can be configured to remain active as RF circuits are periodically switched on when required.
- **Light-sleep mode** – The HP CPU stops running, and can be optionally powered on. The LP peripherals, as well as the LP CPU can be woken up periodically by the timer. The chip can be woken up via all wake up mechanisms: MAC, SDIO host, RTC timer, or external interrupts. Wireless connections can remain active. Some groups of digital peripherals can be optionally powered off.
- **Deep-sleep mode** – Only the LP system is powered on. Wireless connection data is stored in LP memory.

For power consumption in different power modes, see Section 5.6 *Current Consumption Characteristics*.

#### 4.1.3.8 Timer Group

The Timer Group (TIMG) in the ESP32-C6 chip can be used to precisely time an interval, trigger an interrupt after a particular interval (periodically and aperiodically), or act as a hardware clock. ESP32-C6 has two timer groups, each consisting of one general-purpose timer and one Main System Watchdog Timer.

**Feature List**

- 16-bit prescaler
- 54-bit auto-reload-capable up-down counter
- Able to read real-time value of the time-base counter
- Halt, resume, and disable the time-base counter
- Programmable alarm generation
- Timer value reload (auto-reload at an alarm or a software-controlled instant reload)
- RTC slow clock frequency calculation
- Real-time alarm events
- Level interrupt generation
- Support for several ETM tasks and events

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Timer Group (TIMG)*.

#### 4.1.3.9 Watchdog Timers

The Watchdog Timers (WDT) in ESP32-C6 are used to detect and recover from malfunctions. The chip contains three digital watchdog timers: one in each of the two timer groups (MWDT) and one in the RTC Module (RWDT). Additionally, there is one analog watchdog timer called the Super watchdog (SWD) that helps prevent the system from operating in a sub-optimal state.

**Feature List**

- Digital watchdog timers:
  - Four stages, each with a separately programmable timeout value and timeout action
  - Timeout actions: Interrupt, CPU reset, core reset, system reset (RWDT only)
  - Flash boot protection under SPI Boot mode at stage 0
  - Write protection that makes WDT register read only unless unlocked
  - 32-bit timeout counter
- Analog watchdog timer:
  - Timeout period slightly less than one second
  - Timeout actions: Interrupt, system reset

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Watchdog Timers*.

#### 4.1.3.10 Permission Control

The Permission Control module in ESP32-C6 is responsible for managing access permissions to memory and peripheral registers. It consists of two parts: PMP (Physical Memory Protection) and APM (Access Permission Management).

**Feature List**

- Access permission management for ROM, HP memory, HP peripheral, LP memory, and LP peripheral address spaces
- APM supports each master (such as DMA) to select one of the four security modes
- Access permission configuration for up to 16 address ranges
- Interrupt function and exception information record

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Permission Control (PMS)*.

#### 4.1.3.11 System Registers

The System Registers in the ESP32-C6 chip are used to configure various auxiliary chip features.

**Feature List**

- Control External memory encryption and decryption
- Control HP core/LP core debugging

- Control Bus timeout protection

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *System Registers (HP_SYSREG)*.

#### 4.1.3.12 Debug Assistant

The Debug Assistant provides a set of functions to help locate bugs and issues during software debugging. It offers various monitoring capabilities and logging features to assist in identifying and resolving software errors efficiently.

**Feature List**

- Read/write monitoring: Monitor whether the HP CPU bus reads from or writes to a specified memory address space
- Stack pointer (SP) monitoring: Prevent stack overflow or erroneous push/pop operations violation will trigger an interrupt.
- Program counter (PC) logging: Record PC value. The developer can get the last PC value at the most recent HP CPU reset
- Bus access logging: Record information about bus access when the HP CPU, LP CPU, or DMA writes a specified value

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Debug Assistant (ASSIST_DEBUG)*.

### 4.1.4 Cryptography and Security Component

This subsection describes the security features incorporated into the chip, which safeguard data and operations.

#### 4.1.4.1 AES Accelerator

ESP32-C6 integrates an Advanced Encryption Standard (AES) accelerator, which is a hardware device that speeds up computation using AES algorithm significantly, compared to AES algorithms implemented solely in software. The AES accelerator integrated in ESP32-C6 has two working modes, which are Typical AES and DMA-AES.

**Feature List**

- Typical AES working mode
  - AES-128/AES-256 encryption and decryption
- DMA-AES working mode
  - AES-128/AES-256 encryption and decryption
  - Block cipher mode
    - ECB (Electronic Codebook)
    - CBC (Cipher Block Chaining)
    - OFB (Output Feedback)

    * CTR (Counter)
    * CFB8 (8-bit Cipher Feedback)
    * CFB128 (128-bit Cipher Feedback)
  - Interrupt on completion of computation

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *AES Accelerator (AES)*.

#### 4.1.4.2 ECC Accelerator

The ECC Accelerator accelerates calculations based on the Elliptic Curve Cryptography (ECC) algorithm and ECC-derived algorithms like ECDSA, which offers the advantages of smaller public keys compared to RSA cryptography with equivalent security.

**Feature List**

- Supports two different elliptic curves (P-192 and P-256)
- Six working modes that supports Base Point Verification, Base Point Multiplication, Jacobian Point Verification, and Jacobian Point Multiplication

For details, see the *ESP32-C6 Technical Reference Manual* > Chapter *ECC Accelerator (ECC)*.

#### 4.1.4.3 HMAC Accelerator

The HMAC Accelerator (HMAC) module is designed to compute Message Authentication Codes (MACs) using the SHA-256 Hash algorithm and keys as described in RFC 2104. It provides hardware support for HMAC computations, significantly reducing software complexity and improving performance.

**Feature List**

- Standard HMAC-SHA-256 algorithm
- HMAC-SHA-256 calculation based on key in eFuse
  - Whose result cannot be accessed by software in downstream mode for high security
  - Whose result can be accessed by software in upstream mode
- Generates required keys for the Digital Signature Algorithm (DSA) peripheral in downstream mode
- Re-enables soft-disabled JTAG in downstream mode

For details, see the *ESP32-C6 Technical Reference Manual* > Chapter *HMAC Accelerator*.

#### 4.1.4.4 RSA Accelerator

The RSA accelerator provides hardware support for high-precision computation used in various RSA asymmetric cipher algorithms, significantly improving their run time and reducing their software complexity. Compared with RSA algorithms implemented solely in software, this hardware accelerator can speed up RSA algorithms significantly.

**Feature List**

- Large-number modular exponentiation with two optional acceleration options, operands width up to 3072 bits
- Large-number modular multiplication, operands width up to 3072 bits
- Large-number multiplication, operands width up to 1536 bits
- Operands of different widths
- Interrupt on completion of computation

For details, see the *ESP32-C6 Technical Reference Manual* > Chapter *RSA Accelerator*.

#### 4.1.4.5 SHA Accelerator

The SHA Accelerator (SHA) is a hardware device that significantly speeds up the SHA algorithm compared to software-only implementations.

**Feature List**

- Support for multiple SHA algorithms: SHA-1, SHA-224, and SHA-256
- Two working modes: Typical SHA based on CPU and DMA-SHA based on DMA

For more details, see the *ESP32-C6 Technical Reference Manual* > Chapter *SHA Accelerator (SHA)*.

#### 4.1.4.6 Digital Signature

The Digital Signature (DS) module in the ESP32-C6 chip generates message signatures based on RSA with hardware acceleration.

**Feature List**

- RSA digital signatures with key length up to 3072 bits
- Encrypted private key data, only decryptable by DS module
- SHA-256 digest to protect private key data against tampering by an attacker

For more details, see the *ESP32-C6 Technical Reference Manual* > Chapter *Digital Signature (DS)*.

#### 4.1.4.7 External Memory Encryption and Decryption

The External Memory Encryption and Decryption (XTS_AES) module in the ESP32-C6 chip provides security for users' application code and data stored in the external memory (flash).

**Feature List**

- General XTS-AES algorithm, compliant with IEEE Std 1619-2007
- Software-based manual encryption
- High-speed auto decryption without software's participation

- Encryption and decryption functions jointly enabled/disabled by registers configuration, eFuse parameters, and boot mode
- Configurable Anti-DPA

For more details, see the *ESP32-C6 Technical Reference Manual* > Chapter *External Memory Encryption and Decryption (XTS_AES)*.

#### 4.1.4.8 Random Number Generator

The Random Number Generator (RNG) in the ESP32-C6 is a true random number generator that generates 32-bit random numbers for cryptographic operations from a physical process.

**Feature List**

- RNG entropy source
  - Thermal noise from high-speed ADC or SAR ADC
  - An asynchronous clock mismatch

For more details about the Random Number Generator, refer to the *ESP32-C6 Technical Reference Manual* > Chapter *Random Number Generator (RNG)*.

## 4.2 Peripherals

This section describes the chip's peripheral capabilities, covering connectivity interfaces and on-chip sensors that extend its functionality.

### 4.2.1 Connectivity Interface

This subsection describes the connectivity interfaces on the chip that enable communication and interaction with external devices and networks.

#### 4.2.1.1 UART Controller

The UART Controller in the ESP32-C6 chip facilitates the transmission and reception of asynchronous serial data between the chip and external UART devices. It consists of two UARTs in the main system, and one low-power LP UART.

**Feature List**

- Programmable baud rates up to 5 MBaud
- RAM shared by TX FIFOs and RX FIFOs
- Support for various lengths of data bits and stop bits
- Parity bit support
- Special character AT_CMD detection
- RS485 protocol support (not supported by LP UART)
- IrDA protocol support (not supported by LP UART)
- High-speed data communication using GDMA (not supported by LP UART)
- Receive timeout feature
- UART as the wake-up source
- Software and hardware flow control

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *UART Controller (UART, LP_UART)*.

**Pin Assignment**

For UART in the main system, the pins connected to transmit and receive signals (U0TXD and U0RXD) for **UART0** are multiplexed with GPIO16 ~ GPIO17 and FSPICS0 ~ FSPICS1 (chip select for SPI2 interface) via IO MUX. Other signals can be routed to any GPIOs via the GPIO matrix.

For LP UART, the pins used are multiplexed with LP_GPIO0 ~ LP_GPIO5 via LP IO MUX.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.2 SPI Controller

ESP32-C6 has the following SPI interfaces:

- **SPI0** used by ESP32-C6's cache and GDMA to access in-package or off-package flash
- **SPI1** used by the CPU to access in-package or off-package flash
- **SPI2** is a general-purpose SPI controller with access to general-purpose DMA channels

SPI0 and SPI1 are reserved for system use, and only SPI2 is available for users.

#### Features of SPI0 and SPI1

- Supports Single SPI, Dual SPI, Quad SPI (QPI) modes
- Data transmission is in bytes

#### Features of SPI2

- Supports operation as a master or slave
- Support for GDMA
- Supports Single SPI, Dual SPI, Quad SPI (QPI) modes
- Configurable clock polarity (CPOL) and phase (CPHA)
- Configurable clock frequency
- Data transmission is in bytes
- Configurable read and write data bit order: most-significant bit (MSB) first, or least-significant bit (LSB) first
- As a master
  - Supports 2-line full-duplex communication with clock frequency up to 80 MHz
  - Supports 1-, 2-, 4-line half-duplex communication with clock frequency up to 80 MHz
  - Provides six FSPICS… pins for connection with six independent SPI slaves
  - Configurable CS setup time and hold time
- As a slave
  - Supports 2-line full-duplex communication with clock frequency up to 60 MHz
  - Supports 1-, 2-, 4-line half-duplex communication with clock frequency up to 60 MHz

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *SPI Controller (SPI)*.

#### Pin Assignment

For SPI0/1, the pins are multiplexed with GPIO24 ~ GPIO26 and GPIO28 ~ GPIO30 via the IO MUX.

For SPI2, the pins for data and clock signals are multiplexed with GPIO2, GPIO4 ~ GPIO7, and JTAG interface via the IO MUX. The pins for chip select signals for multiplexed with GPIO16 ~ GPIO21, UART0 interface, and SDIO interface via the IO MUX.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.3 I2C Controller

The I2C Controller supports communication between the master and slave devices using the I2C bus.

**Feature List**

- Two I2C controllers: one in the main system and one in the low-power system
- Communication with multiple external devices
- Master and slave modes for I2C, and master mode only for LP I2C
- Standard mode (100 Kbit/s) and fast mode (400 Kbit/s)
- SCL clock stretching in slave mode
- Programmable digital noise filtering
- Support for 7-bit and 10-bit addressing, as well as dual address mode

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *I2C Controller (I2C)*.

**Pin Assignment**

For regular I2C, the pins used can be chosen from any GPIOs via the GPIO Matrix.

For LP I2C, the pins used are multiplexed with LP_GPIO6 ~ LP_GPIO7 via LP IO MUX.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.4 I2S Controller

The I2S Controller in the ESP32-C6 chip provides a flexible communication interface for streaming digital data in multimedia applications, particularly digital audio applications.

**Feature List**

- Master mode and slave mode
- Full-duplex and half-duplex communications
- Separate TX and RX units that can work independently or simultaneously
- A variety of audio standards supported:
  - TDM Philips standard
  - TDM MSB alignment standard
  - TDM PCM standard
  - PDM standard
- PCM-to-PDM TX interface
- Configurable high-precision BCK clock, with frequency up to 40 MHz
  - Sampling frequencies can be 8 kHz, 16 kHz, 32 kHz, 44.1 kHz, 48 kHz, 88.2 kHz, 96 kHz, 128 kHz, 192 kHz, etc.

- 8-/16-/24-/32-bit data communication
- Direct Memory Access (DMA)
- A-law and μ-law compression/decompression algorithms for improved signal-to-quantization noise ratio
- Flexible data format control

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *I2S Controller (I2S)*.

**Pin Assignment**

The pins for the I2S Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.5 Pulse Count Controller

The Pulse Count Controller (PCNT) is designed to count input pulses by tracking rising and falling edges of the input pulse signal.

**Feature List**

- Four independent pulse counters with two channels each
- Counter modes: increment, decrement, or disable
- Glitch filtering for input pulse signals and control signals
- Selection between counting on rising or falling edges of the input pulse signal

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Pulse Count Controller*.

**Pin Assignment**

The pins for the Pulse Count Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.6 USB Serial/JTAG Controller

The USB Serial/JTAG controller in the ESP32-C6 chip provides an integrated solution for communicating to the chip over a standard USB CDC-ACM serial port as well as a convenient method for JTAG debugging. It eliminates the need for external chips or JTAG adapters, saving space and reducing cost.

**Feature List**

- USB 2.0 full speed compliant, capable of up to 12 Mbit/s transfer speed (Note that this controller does not support the faster 480 Mbit/s high-speed transfer mode)
- CDC-ACM virtual serial port and JTAG adapter functionality
- CDC-ACM:

  - CDC-ACM adherent serial port emulation (plug-and-play on most modern OSes)
  - Host controllable chip reset and entry into download mode
- JTAG adapter functionality:
  - Fast communication with CPU debugging core using a compact representation of JTAG instructions
- Support for reprogramming of attached flash memory through the ROM startup code
- Internal PHY

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *USB Serial/JTAG Controller (USB_SERIAL_JTAG)*.

**Pin Assignment**

The pins for the USB Serial/JTAG Controller are multiplexed with GPIO12 ~ GPIO13.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.7 Two-wire Automotive Interface

The Two-wire Automotive Interface (TWAI®) is a multi-master, multi-cast communication protocol designed for automotive applications. The TWAI controller facilitates the communication based on this protocol.

**Feature List**

- Compatible with ISO 11898-1 protocol (CAN Specification 2.0)
- Standard frame format (11-bit ID) and extended frame format (29-bit ID)
- Bit rates from 1 Kbit/s to 1 Mbit/s
- Multiple modes of operation: Normal, Listen Only, and Self-Test (no acknowledgment required)
- Special transmissions: Single-shot and Self Reception
- Acceptance filter (single and dual filter modes)
- Error detection and handling: error counters, configurable error warning limit, error code capture, arbitration lost capture, automatic transceiver standby

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Two-wire Automotive Interface*.

**Pin Assignment**

The pins for the Two-wire Automotive Interface can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.8 SDIO 2.0 Slave Controller

The SDIO 2.0 Slave Controller in the ESP32-C6 chip provides hardware support for the Secure Digital Input/Output (SDIO) device interface. It allows an SDIO host to access the ESP32-C6 via an SDIO bus

protocol.

**Feature List**

- Compatible with SD Physical Layer Specification V2.00 and SDIO V2.00 specifications
- Support for SPI, 1-bit SDIO, and 4-bit SDIO transfer modes
- Clock range of 0 ~ 50 MHz
- Configurable sample and drive clock edge
- Integrated and SDIO-accessible registers for information interaction
- Support for SDIO interrupt mechanism
- Automatic padding data and discarding the padded data on the SDIO bus
- Block size up to 512 bytes
- Interrupt vector between the host and slave for bidirectional interrupt
- Support DMA for data transfer
- Support for wake-up from sleep when connection is retained

For more details about the SDIO 2.0 Slave Controller, refer to the *ESP32-C6 Technical Reference Manual* > Chapter *SDIO 2.0 Slave Controller (SDIO)*.

**Pin Assignment**

The pins for the SDIO 2.0 Slave Controller are multiplexed with GPIO18 ~ GPIO23 and FSPICS2 ~ FSPICS5 via IO MUX.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.9 LED PWM Controller

The LED PWM Controller (LEDC) is designed to generate PWM signals for LED control.

**Feature List**

- Six independent PWM generators
- Maximum PWM duty cycle resolution of 20 bits
- Four independent timers with 20-bit counters, configurable fractional clock dividers and counter overflow values
- Adjustable phase of PWM signal output
- PWM duty cycle dithering
- Automatic duty cycle fading
  - Linear duty cycle fading — only one duty cycle range

  - Gamma curve fading — up to 16 duty cycle ranges for each PWM generator, with independently configured fading direction (increase or decrease), fading amount, number of fades, and fading frequency
- PWM signal output in low-power mode (Light-sleep mode)
- Event generation and task response achieved by the Event Task Matrix (ETM)

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *LED PWM Controller*.

**Pin Assignment**

The pins for the LED PWM Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.10 Motor Control PWM

The Motor Control Pulse Width Modulator (MCPWM) is designed for driving digital motors and smart light. The MCPWM is divided into five main modules: PWM timers, PWM operators, Capture module, Fault Detection module, and Event Task Matrix (ETM) module.

**Feature List**

- Three PWM timers for precise timing and frequency control
  - Every PWM timer has a dedicated 8-bit clock prescaler
  - The 16-bit counter in the PWM timer can work in count-up mode, count-down mode, or count-up-down mode
  - Hardware or software synchronization to trigger a reload on the PWM timer or the prescaler's restart, with selectable hardware synchronization source
- Three PWM operators for generating waveform pairs
  - Six PWM outputs to operate in several topologies
  - Configurable dead time on rising and falling edges; each set up independently
  - Modulating of PWM output by high-frequency carrier signals, useful when gate drivers are insulated with a transformer
- Capture module for hardware-based signal processing
  - Speed measurement of rotating machinery
  - Measurement of elapsed time between position sensor pulses
  - Period and duty cycle measurement of pulse train signals
  - Decoding current or voltage amplitude derived from duty-cycle-encoded signals of current/voltage sensors
  - Three individual capture channels, each of which with a 32-bit time-stamp register
  - Selection of edge polarity and prescaling of input capture signals

  - The capture timer can sync with a PWM timer or external signals
- Fault Detection module
  - Programmable fault handling in both cycle-by-cycle mode and one-shot mode
  - A fault condition can force the PWM output to either high or low logic levels
- Event generation and task response achieved by the Event Task Matrix (ETM)

For details, see *ESP32-C6 Technical Reference Manual* > Chapter *Motor Control PWM (MCPWM)*.

**Pin Assignment**

The pins for the Motor Control PWM can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.11 Remote Control Peripheral

The Remote Control Peripheral (RMT) controls the transmission and reception of infrared remote control signals.

**Feature List**

- Four channels for sending and receiving infrared remote control signals
- Independent transmission and reception capabilities for each channel
- Support for Normal TX/RX mode, Wrap TX/RX mode, Continuous TX mode
- Modulation on TX pulses and Demodulation on RX pulses
- RX filtering for improved signal reception
- Ability to transmit data simultaneously on multiple channels
- Clock divider counter, state machine, and receiver for each RX channel
- Default allocation of RAM blocks to channels based on channel number
- RAM containing 16-bit entries with “level” and “period” fields

For more details, see *ESP32-C6 Technical Reference Manual* > Chapter *Remote Control Peripheral (RMT)*.

**Pin Assignment**

The pins for the Remote Control Peripheral can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.1.12 Parallel IO Controller

The Parallel IO Controller (PARLIO) in the ESP32-C6 chip enables data transfer between external devices and internal memory on a parallel bus through GDMA. It consists of a transmitter (TX unit) and a receiver (RX unit), making it a versatile interface for connecting various peripherals.

**Feature List**

- 1/2/4/8/16-bit configurable data bus width
- Half-duplex communication with 16-bit data bus width and full-duplex communication with 8-bit data bus width
- Bit reordering in 1/2/4-bit data bus width mode
- RX unit supports 15 receive modes categorized into three major categories: Level Enable mode, Pulse Enable mode, and Software Enable mode
- TX unit can generate a valid signal aligned with TX

For more details, see *ESP32-C6 Technical Reference Manual* > Chapter *Parallel IO Controller*.

**Pin Assignment**

The pins for the Parallel IO Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

### 4.2.2 Analog Signal Processing

This subsection describes components on the chip that sense and process real-world data.

#### 4.2.2.1 SAR ADC

ESP32-C6 integrates a Successive Approximation Analog-to-Digital Converter (SAR ADC) to convert analog signals into digital representations.

**Feature List**

- 12-bit sampling resolution
- Analog voltage sampling from up to seven pins
- Attenuation of input signals for voltage conversion
- Software-triggered one-time sampling
- Timer-triggered multi-channel scanning
- DMA continuous conversion for seamless data transfer
- Two filters with configurable filter coefficient
- Threshold monitoring which helps to trigger an interrupt
- Support for Event Task Matrix

For more details, see *ESP32-C6 Technical Reference Manual* > Chapter *On-Chip Sensors and Analog Signal Processing*.

**Pin Assignment**

The pins for the SAR ADC are multiplexed with GPIO0 ~ GPIO6, LP_GPIO0 ~ LP_GPIO6, JTAG interface, SPI2 interface, LP UART interface, and LP I2C interface.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-C6 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

#### 4.2.2.2 Temperature Sensor

The Temperature Sensor in the ESP32-C6 chip allows for real-time monitoring of temperature changes inside the chip.

**Feature List**

- Measurement range: –40°C ~ 125°C
- Software triggering, wherein the data can be read continuously once triggered
- Hardware automatic triggering and temperature monitoring
- Configurable temperature offset based on the environment to improve the accuracy
- Adjustable measurement range
- Two automatic monitoring wake-up modes: absolute value mode and incremental value mode
- Support for Event Task Matrix

For more details, see *ESP32-C6 Technical Reference Manual* > Chapter *On-Chip Sensors and Analog Signal Processing*.

## 4.3 Wireless Communication

This section describes the chip's wireless communication capabilities, spanning radio technology, Wi-Fi, Bluetooth, and 802.15.4.

### 4.3.1 Radio

This subsection describes the fundamental radio technology embedded in the chip that facilitates wireless communication and data exchange. The ESP32-C6 radio consists of the following blocks:

- 2.4 GHz receiver
- 2.4 GHz transmitter
- bias and regulators
- balun and transmit-receive switch
- clock generator

#### 4.3.1.1 2.4 GHz Receiver

The 2.4 GHz receiver demodulates the 2.4 GHz RF signal to quadrature baseband signals and converts them to the digital domain with two high-resolution, high-speed ADCs. To adapt to varying signal channel conditions, ESP32-C6 integrates RF filters, Automatic Gain Control (AGC), DC offset cancelation circuits, and baseband filters.

#### 4.3.1.2 2.4 GHz Transmitter

The 2.4 GHz transmitter modulates the quadrature baseband signals to the 2.4 GHz RF signal, and drives the antenna with a high-powered CMOS power amplifier. The use of digital calibration further improves the linearity of the power amplifier.

Additional calibrations are integrated to cancel any radio imperfections, such as:

- carrier leakage
- I/Q amplitude/phase matching
- baseband nonlinearities
- RF nonlinearities
- antenna matching

These built-in calibration routines reduce the cost, time, and specialized equipment required for product testing.

#### 4.3.1.3 Clock Generator

The clock generator produces quadrature clock signals of 2.4 GHz for both the receiver and the transmitter. All components of the clock generator are integrated into the chip, including inductors, varactors, filters, regulators and dividers.

The clock generator has built-in calibration and self-test circuits. Quadrature clock phases and phase noise are optimized on chip with patented calibration algorithms which ensure the best performance of the receiver and the transmitter.

### 4.3.2 Wi-Fi

This subsection describes the chip's Wi-Fi capabilities, which facilitate wireless communication at a high data rate.

#### 4.3.2.1 Wi-Fi Radio and Baseband

The ESP32-C6 Wi-Fi radio and baseband support the following features:

- compliant with IEEE 802.11b/g/n/ax
- 1T1R in 2.4 GHz band
- 802.11ax
  - 20 MHz-only non-AP mode
  - MCS0 ~MCS9
  - uplink and downlink OFDMA
  - downlink MU-MIMO (multi-user, multiple input, multiple output)
  - longer OFDM symbol, with 0.8, 1.6, 3.2 $\mu$s guard interval
  - DCM (dual carrier modulation), up to 16-QAM
  - single-user/multi-user beamformee
  - channel quality indication (CQI)
  - RX STBC (single spatial stream)
- 802.11b/g/n
  - MCS0 ~MCS7 that supports 20 MHz and 40 MHz bandwidth
  - MCS32
  - data rate up to 150 Mbps
  - 0.4 $\mu$s guard interval
- adjustable transmitting power
- antenna diversity
  ESP32-C6 supports antenna diversity with an external RF switch. This switch is controlled by one or more GPIOs, and used to select the best antenna to minimize the effects of channel imperfections.

#### 4.3.2.2 Wi-Fi MAC

ESP32-C6 implements the full IEEE 802.11 b/g/n/ax Wi-Fi MAC protocol. It supports the Basic Service Set (BSS) STA and SoftAP operations under the Distributed Control Function (DCF). Power management is handled automatically with minimal host interaction to minimize the active duty period.

The ESP32-C6 Wi-Fi MAC applies the following low-level protocol functions automatically:

- 4 × virtual Wi-Fi interfaces
- infrastructure BSS in Station mode, SoftAP mode, Station + SoftAP mode, and promiscuous mode
- RTS protection, CTS protection, Immediate Block ACK
- fragmentation and defragmentation
- TX/RX A-MPDU, TX/RX A-MSDU
- transmit opportunity (TXOP)
- Wi-Fi multimedia (WMM)
- GCMP, CCMP, TKIP, WAPI, WEP, BIP, WPA2-PSK/WPA2-Enterprise, and WPA3-PSK/WPA3-Enterprise
- automatic beacon monitoring (hardware TSF)
- 802.11mc FTM
- 802.11ax supports:
  - target wake time (TWT) requester
  - multiple BSSIDs
  - triggered response scheduling
  - uplink power headroom
  - operating mode
  - buffer status report
  - Multi-user Request-to-Send (MU-RTS), Multi-user Block ACK Request (MU-BAR), and Multi-STA Block ACK (M-BA) frame
  - intra-PPDU power saving mechanism
  - two network allocation vectors (NAV)
  - BSS coloring
  - spatial reuse
  - uplink power headroom
  - operating mode control
  - buffer status report
  - TXOP duration RTS threshold
  - UL-OFDMA random access (UORA)

#### 4.3.2.3 Networking Features

Espressif provides libraries for TCP/IP networking, ESP-WIFI-MESH networking, and other networking protocols over Wi-Fi. TLS 1.0, 1.1 and 1.2 is also supported.

### 4.3.3 Bluetooth LE

This subsection describes the chip's Bluetooth capabilities, which facilitate wireless communication for low-power, short-range applications. ESP32-C6 includes a Bluetooth Low Energy subsystem that integrates a hardware link controller, an RF/modem block and a feature-rich software protocol stack. It supports the core features of Bluetooth 5 and Bluetooth mesh.

#### 4.3.3.1 Bluetooth LE PHY

Bluetooth Low Energy PHY in ESP32-C6 supports:

- 1 Mbps PHY
- 2 Mbps PHY for higher data rates
- coded PHY for longer range (125 Kbps and 500 Kbps)
- HW listen before talk (LBT)

#### 4.3.3.2 Bluetooth LE Link Controller

Bluetooth Low Energy Link Controller in ESP32-C6 supports:

- LE advertising extensions, to enhance broadcasting capacity and broadcast more intelligent data
- multiple advertisement sets
- simultaneous advertising and scanning
- multiple connections in simultaneous central and peripheral roles
- adaptive frequency hopping and channel assessment
- LE channel selection algorithm #2
- LE power control
- connection parameter update
- high duty cycle non-connectable advertising
- LE privacy 1.2
- LE data packet length extension
- link layer extended scanner filter policies
- low duty cycle directed advertising
- link layer encryption
- LE Ping

### 4.3.4 802.15.4

This subsection describes the chip's compatibility with the 802.15.4 standard, which facilitates wireless communication for low-power, short-range applications. ESP32-C6 includes an IEEE Standard 802.15.4 subsystem that integrates PHY and MAC layer. It supports various software stacks including Thread, Zigbee, Matter, HomeKit, MQTT and so on.

#### 4.3.4.1 802.15.4 PHY

ESP32-C6 's 802.15.4 PHY supports:

- O-QPSK PHY in 2.4 GHz
- 250 Kbps data rate
- RSSI and LQI supported

#### 4.3.4.2 802.15.4 MAC

ESP32-C6 supports most key features defined in IEEE Standard 802.15.4-2015, including:

- CSMA/CA
- active scan and energy detect
- HW frame filter
- HW auto acknowledge
- HW auto frame pending
- coordinated sampled listening (CSL)

# 5 Electrical Characteristics

## 5.1 Absolute Maximum Ratings

Stresses above those listed in Table 5-1 *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only and normal operation of the device at these or any other conditions beyond those indicated in Section 5.2 *Recommended Operating Conditions* is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

**Table 5-1. Absolute Maximum Ratings**

| Parameter | Description | Min | Max | Unit |
|---|---|---|---|---|
| Input power pins[1] | Allowed input voltage | –0.3 | 3.6 | V |
| $I_{output}$[2] | Cumulative IO output current | — | 1000 | mA |
| $T_{STORE}$ | Storage temperature | –40 | 150 | °C |

[1] For more information on input power pins, see Section 2.5.1 *Power Pins*.

[2] The product proved to be fully functional after all its IO pins were pulled high while being connected to ground for 24 consecutive hours at ambient temperature of 25 °C.

## 5.2 Recommended Operating Conditions

**Table 5-2. Recommended Power Characteristics**

| Parameter [1] | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| VDDA1, VDDA2, VDDA3P3 | Recommended input voltage | 3.0 | 3.3 | 3.6 | V |
| VDDPST1 | Recommended input voltage | 3.0 | 3.3 | 3.6 | V |
| VDD_SPI (as input) | — | 3.0 | 3.3 | 3.6 | V |
| VDDPST2 [2, 3] | Recommended input voltage | 3.0 | 3.3 | 3.6 | V |
| $I_{VDD}$ | Cumulative input current | 0.5 | — | — | A |
| $T_A$ | Ambient temperature | –40 | — | 105 | °C |

[1] See in conjunction with Section 2.5 *Power Supply*.

[2] If VDDPST2 is used to power VDD_SPI (see Section 2.5.2 *Power Scheme*), the voltage drop on $R_{SPI}$ should be accounted for. See also Section 5.3 *VDD_SPI Output Characteristics*.

[3] If writing to eFuses, the voltage on VDDPST2 should not exceed 3.3 V as the circuits responsible for burning eFuses are sensitive to higher voltages.

## 5.3 VDD_SPI Output Characteristics

**Table 5-3. VDD_SPI Internal and Output Characteristics**

| Parameter | Description [1] | Typ | Unit |
|---|---|---|---|
| $R_{SPI}$ | VDD_SPI powered by VDDPST2 via $R_{SPI}$ for 3.3 V flash [2] | 3 | Ω |

[1] See in conjunction with Section 2.5.2 *Power Scheme*.

[2] VDD3P3_RTC must be more than *VDD_flash_min* + *I_flash_max* * $R_{SPI}$; where

- *VDD_flash_min* – minimum operating voltage of flash
- *I_flash_max* – maximum operating current of flash

## 5.4 DC Characteristics (3.3 V, 25 °C)

**Table 5-4. DC Characteristics (3.3 V, 25 °C)**

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $C_{IN}$ | Pin capacitance | — | 2 | — | pF |
| $V_{IH}$ | High-level input voltage | 0.75 × VDD [1] | — | VDD [1] + 0.3 | V |
| $V_{IL}$ | Low-level input voltage | –0.3 | — | 0.25 × VDD [1] | V |
| $I_{IH}$ | High-level input current | — | — | 50 | nA |
| $I_{IL}$ | Low-level input current | — | — | 50 | nA |
| $V_{OH}$ [2] | High-level output voltage | 0.8 × VDD [1] | — | — | V |
| $V_{OL}$ [2] | Low-level output voltage | — | — | 0.1 × VDD [1] | V |
| $I_{OH}$ | High-level source current (VDD [1] = 3.3 V, $V_{OH}$ >= 2.64 V, PAD_DRIVER = 3) | — | 40 | — | mA |
| $I_{OL}$ | Low-level sink current (VDD [1] = 3.3 V, $V_{OL}$ = 0.495 V, PAD_DRIVER = 3) | — | 28 | — | mA |
| $R_{PU}$ | Internal weak pull-up resistor | — | 45 | — | kΩ |
| $R_{PD}$ | Internal weak pull-down resistor | — | 45 | — | kΩ |
| $V_{IH\_nRST}$ | Chip reset release voltage CHIP_PU voltage is within the specified range) | 0.75 × VDD [1] | — | VDD [1] + 0.3 | V |
| $V_{IL\_nRST}$ | Chip reset voltage (CHIP_PU voltage is within the specified range) | –0.3 | — | 0.25 × VDD [1] | V |

[1] VDD – voltage from a power pin of a respective power domain.

[2] $V_{OH}$ and $V_{OL}$ are measured using high-impedance load.

## 5.5 ADC Characteristics

The measurements in this section are taken with an external 100 nF capacitor connected to the ADC, using DC signals as input, and at an ambient temperature of 25 °C with disabled Wi-Fi.

**Table 5-5. ADC Characteristics**

| Symbol | Min | Max | Unit |
|---|---|---|---|
| DNL (Differential nonlinearity) [1] | –8 | 12 | LSB |
| INL (Integral nonlinearity) | –10 | 10 | LSB |
| Sampling rate | — | 100 | kSPS [2] |

[1] To get better DNL results, you can sample multiple times and apply a filter, or calculate the average value.

[2] kSPS means kilo samples-per-second.

The calibrated ADC results after hardware calibration and software calibration are shown in Table 5-6. For higher accuracy, you may implement your own calibration methods.

**Table 5-6. ADC Calibration Results**

| Parameter | Description | Min | Max | Unit |
|---|---|---|---|---|
| Total error | ATTEN0, effective measurement range of 0 ~ 1000 | –12 | 12 | mV |
| | ATTEN1, effective measurement range of 0 ~ 1300 | –12 | 12 | mV |
| | ATTEN2, effective measurement range of 0 ~ 1900 | –23 | 23 | mV |
| | ATTEN3, effective measurement range of 0 ~ 3300 | –40 | 40 | mV |

**Note:**

The above ADC measurement range and accuracy are applicable to chips manufactured on and after the Date Code **212023** on shielding cases, or assembled on and after the D/C 1 and D/C 2 **2321** on bar-code labels. For chips manufactured or assembled earlier than these date codes, please ask our sales team to provide the actual range and accuracy according to batch.

For details of Date Code and D/C, please refer to *Espressif Chip Packaging Information*.

## 5.6 Current Consumption Characteristics

### 5.6.1 Current Consumption in Active Mode

The current consumption measurements are taken with a 3.3 V supply at 25 °C ambient temperature.

TX current consumption is rated at a 100% duty cycle.

RX current consumption is rated when the peripherals are disabled and the CPU idle.

**Table 5-7. Current Consumption for Wi-Fi (2.4 GHz) in Active Mode**

| Work Mode | RF Condition | Description | Peak (mA) |
|---|---|---|---|
| Active (RF working) | TX | 802.11b, 1 Mbps, DSSS @ 21.0 dBm | 354 |
| | | 802.11g, 54 Mbps, OFDM @ 19.5 dBm | 300 |
| | | 802.11n, HT20, MCS7 @ 18.5 dBm | 280 |
| | | 802.11n, HT40, MCS7 @ 18.0 dBm | 268 |
| | | 802.11ax, MCS9, @ 16.5 dBm | 252 |

**Cont'd on next page**

Table 5-7 – cont'd from previous page

| Work Mode | RF Condition | Description | Peak (mA) |
|---|---|---|---|
| | RX | 802.11b/g/n, HT20 | 78 |
| | | 802.11n, HT40 | 82 |
| | | 802.11ax, HE20 | 78 |

Table 5-8. Current Consumption for Bluetooth LE in Active Mode

| Work Mode | RF Condition | Description | Peak (mA) |
|---|---|---|---|
| Active (RF working) | TX | Bluetooth LE @ 20.0 dBm | 315 |
| | | Bluetooth LE @ 9.0 dBm | 190 |
| | | Bluetooth LE @ 0 dBm | 130 |
| | | Bluetooth LE @ –15.0 dBm | 94 |
| | RX | Bluetooth LE | 71 |

Table 5-9. Current Consumption for 802.15.4 in Active Mode

| Work Mode | RF Condition | Description | Peak (mA) |
|---|---|---|---|
| Active (RF working) | TX | 802.15.4 @ 20.0 dBm | 305 |
| | | 802.15.4 @ 12.0 dBm | 187 |
| | | 802.15.4 @ 0 dBm | 119 |
| | | 802.15.4 @ –15.0 dBm | 92 |
| | RX | 802.15.4 | 74 |

### 5.6.2 Current Consumption in Other Modes

Table 5-10. Current Consumption in Modem-sleep Mode

| Mode | CPU Frequency (MHz) | Description | Typ (mA) | |
|---|---|---|---|---|
| | | | All Peripherals Clocks Disabled | All Peripherals Clocks Enabled [1] |
| Modem-sleep [2,3] | 160 | CPU is running | 27 | 38 |
| | | CPU is idle | 17 | 28 |
| | 80 | CPU is running | 19 | 30 |
| | | CPU is idle | 14 | 25 |

[1] In practice, the current consumption might be different depending on which peripherals are enabled.

[2] In Modem-sleep mode, Wi-Fi is clock gated.

[3] In Modem-sleep mode, the consumption might be higher when accessing flash.

**Table 5-11. Current Consumption in Low-Power Modes**

| Mode | Description | Typ ($\mu$A) |
|---|---|---|
| Light-sleep | CPU and wireless communication modules are powered down, peripheral clocks are disabled, and all GPIOs are high-impedance | 180 |
| | CPU, wireless communication modules and peripherals are powered down, and all GPIOs are high-impedance | 35 |
| Deep-sleep | RTC timer and LP memory are powered on | 7 |
| Power off | CHIP_PU is set to low level, the chip is powered off | 1 |

## 5.7 Reliability

**Table 5-12. Reliability Qualifications**

| Test Item | Test Conditions | Test Standard |
|---|---|---|
| HTOL (High Temperature Operating Life) | 125 °C, 1000 hours | JESD22-A108 |
| ESD (Electro-Static Discharge Sensitivity) | HBM (Human Body Mode) [1] ± 2000 V | JS-001 |
| | CDM (Charge Device Mode) [2] ± 1000 V | JS-002 |
| Latch up | Current trigger ± 200 mA | JESD78 |
| | Voltage trigger 1.5 × $VDD_{max}$ | |
| Preconditioning | Bake 24 hours @125 °C<br>Moisture soak (level 3: 192 hours @30 °C, 60% RH)<br>IR reflow solder: 260 + 0 °C, 20 seconds, three times | J-STD-020, JESD47, JESD22-A113 |
| TCT (Temperature Cycling Test) | –65 °C / 150 °C, 500 cycles | JESD22-A104 |
| uHAST (Highly Accelerated Stress Test, unbiased) | 130 °C, 85% RH, 96 hours | JESD22-A118 |
| HTSL (High Temperature Storage Life) | 150 °C, 1000 hours | JESD22-A103 |
| LTSL (Low Temperature Storage Life) | –40 °C, 1000 hours | JESD22-A119 |

[1] JEDEC document JEP155 states that 500 V HBM allows safe manufacturing with a standard ESD control process.

[2] JEDEC document JEP157 states that 250 V CDM allows safe manufacturing with a standard ESD control process.

# 6 RF Characteristics

This section contains tables with RF characteristics of the Espressif product.

The RF data is measured at the antenna port, where RF cable is connected, including the front-end loss. The front-end circuit is a 0 Ω resistor.

Devices should operate in the center frequency range allocated by regional regulatory authorities. The target center frequency range and the target transmit power are configurable by software. See ESP RF Test Tool and Test Guide for instructions.

Unless otherwise stated, the RF tests are conducted with a 3.3 V (±5%) supply at 25 °C ambient temperature.

## 6.1 Wi-Fi Radio

**Table 6-1. Wi-Fi RF Characteristics**

| Name | Description |
|---|---|
| Center frequency range of operating channel | 2412 ~ 2484 MHz |
| Wi-Fi wireless standard | IEEE 802.11b/g/n/ax |

### 6.1.1 Wi-Fi RF Transmitter (TX) Characteristics

**Table 6-2. TX Power with Spectral Mask and EVM Meeting 802.11 Standards**

| Rate | Min (dBm) | Typ (dBm) | Max (dBm) |
|---|---|---|---|
| 802.11b, 1 Mbps, DSSS | — | 21.0 | — |
| 802.11b, 11 Mbps, CCK | — | 21.0 | — |
| 802.11g, 6 Mbps, OFDM | — | 20.5 | — |
| 802.11g, 54 Mbps, OFDM | — | 19.5 | — |
| 802.11n, HT20, MCS0 | — | 19.5 | — |
| 802.11n, HT20, MCS7 | — | 18.5 | — |
| 802.11n, HT40, MCS0 | — | 19.0 | — |
| 802.11n, HT40, MCS7 | — | 18.0 | — |
| 802.11ax, HE20, MCS0 | — | 19.5 | — |
| 802.11ax, HE20, MCS9 | — | 16.5 | — |

**Table 6-3. TX EVM Test**[1]

| Rate | Min (dB) | Typ (dB) | Limit (dB) |
|---|---|---|---|
| 802.11b, 1 Mbps, DSSS | — | –25.5 | –10.0 |
| 802.11b, 11 Mbps, CCK | — | –25.5 | –10.0 |
| 802.11g, 6 Mbps, OFDM | — | –26.5 | –5.0 |

**Cont'd on next page**

Table 6-3 – cont'd from previous page

| Rate | Min (dB) | Typ (dB) | Limit (dB) |
|---|---|---|---|
| 802.11g, 54 Mbps, OFDM | — | –29.0 | –25.0 |
| 802.11n, HT20, MCS0 | — | –29.0 | –5.0 |
| 802.11n, HT20, MCS7 | — | –30.0 | –27.0 |
| 802.11n, HT40, MCS0 | — | –28.5 | –5.0 |
| 802.11n, HT40, MCS7 | — | –29.5 | –27.0 |
| 802.11ax, HE20, MCS0 | — | –29.0 | –5.0 |
| 802.11ax, HE20, MCS9 | — | –34.0 | –32.0 |

[1] EVM is measured at the corresponding typical TX power provided in Table 6-2 *TX Power with Spectral Mask and EVM Meeting 802.11 Standards* above.

### 6.1.2 Wi-Fi RF Receiver (RX) Characteristics

For RX tests, the PER (packet error rate) limit is 8% for 802.11b, and 10% for 802.11g/n/ax.

Table 6-4. RX Sensitivity

| Rate | Min (dBm) | Typ (dBm) | Max (dBm) |
|---|---|---|---|
| 802.11b, 1 Mbps, DSSS | — | –99.2 | — |
| 802.11b, 2 Mbps, DSSS | — | –96.8 | — |
| 802.11b, 5.5 Mbps, CCK | — | –93.8 | — |
| 802.11b, 11 Mbps, CCK | — | –90.0 | — |
| 802.11g, 6 Mbps, OFDM | — | –94.0 | — |
| 802.11g, 9 Mbps, OFDM | — | –93.2 | — |
| 802.11g, 12 Mbps, OFDM | — | –92.6 | — |
| 802.11g, 18 Mbps, OFDM | — | –90.0 | — |
| 802.11g, 24 Mbps, OFDM | — | –86.8 | — |
| 802.11g, 36 Mbps, OFDM | — | –83.2 | — |
| 802.11g, 48 Mbps, OFDM | — | –79.0 | — |
| 802.11g, 54 Mbps, OFDM | — | –77.6 | — |
| 802.11n, HT20, MCS0 | — | –93.6 | — |
| 802.11n, HT20, MCS1 | — | –92.4 | — |
| 802.11n, HT20, MCS2 | — | –89.6 | — |
| 802.11n, HT20, MCS3 | — | –86.2 | — |
| 802.11n, HT20, MCS4 | — | –82.8 | — |
| 802.11n, HT20, MCS5 | — | –78.8 | — |
| 802.11n, HT20, MCS6 | — | –77.2 | — |
| 802.11n, HT20, MCS7 | — | –75.6 | — |
| 802.11n, HT40, MCS0 | — | –91.0 | — |
| 802.11n, HT40, MCS1 | — | –90.0 | — |
| 802.11n, HT40, MCS2 | — | –87.4 | — |

Cont'd on next page

Table 6-4 – cont'd from previous page

| Rate | Min (dBm) | Typ (dBm) | Max (dBm) |
|---|---|---|---|
| 802.11n, HT40, MCS3 | — | –83.8 | — |
| 802.11n, HT40, MCS4 | — | –80.8 | — |
| 802.11n, HT40, MCS5 | — | –76.6 | — |
| 802.11n, HT40, MCS6 | — | –75.0 | — |
| 802.11n, HT40, MCS7 | — | –73.4 | — |
| 802.11ax, HE20, MCS0 | — | –93.8 | — |
| 802.11ax, HE20, MCS1 | — | –91.2 | — |
| 802.11ax, HE20, MCS2 | — | –88.4 | — |
| 802.11ax, HE20, MCS3 | — | –85.6 | — |
| 802.11ax, HE20, MCS4 | — | –82.2 | — |
| 802.11ax, HE20, MCS5 | — | –78.4 | — |
| 802.11ax, HE20, MCS6 | — | –76.6 | — |
| 802.11ax, HE20, MCS7 | — | –74.8 | — |
| 802.11ax, HE20, MCS8 | — | –71.0 | — |
| 802.11ax, HE20, MCS9 | — | –69.0 | — |

Table 6-5. Maximum RX Level

| Rate | Min (dBm) | Typ (dBm) | Max (dBm) |
|---|---|---|---|
| 802.11b, 1 Mbps, DSSS | — | 5 | — |
| 802.11b, 11 Mbps, CCK | — | 5 | — |
| 802.11g, 6 Mbps, OFDM | — | 5 | — |
| 802.11g, 54 Mbps, OFDM | — | 0 | — |
| 802.11n, HT20, MCS0 | — | 5 | — |
| 802.11n, HT20, MCS7 | — | 0 | — |
| 802.11n, HT40, MCS0 | — | 5 | — |
| 802.11n, HT40, MCS7 | — | 0 | — |
| 802.11ax, HE20, MCS0 | — | 5 | — |
| 802.11ax, HE20, MCS9 | — | 0 | — |

Table 6-6. RX Adjacent Channel Rejection

| Rate | Min (dB) | Typ (dB) | Max (dB) |
|---|---|---|---|
| 802.11b, 1 Mbps, DSSS | — | 38 | — |
| 802.11b, 11 Mbps, CCK | — | 38 | — |
| 802.11g, 6 Mbps, OFDM | — | 31 | — |
| 802.11g, 54 Mbps, OFDM | — | 20 | — |
| 802.11n, HT20, MCS0 | — | 31 | — |

Cont'd on next page

Table 6-6 – cont'd from previous page

| Rate | Min (dB) | Typ (dB) | Max (dB) |
|---|---|---|---|
| 802.11n, HT20, MCS7 | — | 16 | — |
| 802.11n, HT40, MCS0 | — | 28 | — |
| 802.11n, HT40, MCS7 | — | 10 | — |
| 802.11ax, HE20, MCS0 | — | 25 | — |
| 802.11ax, HE20, MCS9 | — | 2 | — |

## 6.2 Bluetooth 5 (LE) Radio

Table 6-7. Bluetooth LE RF Characteristics

| Name | Description |
|---|---|
| Center frequency range of operating channel | 2402 ~ 2480 MHz |
| RF transmit power range | –15.0 ~ 20.0 dBm |

### 6.2.1 Bluetooth LE RF Transmitter (TX) Characteristics

Table 6-8. Bluetooth LE - Transmitter Characteristics - 1 Mbps

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Carrier frequency offset and drift | Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$ | — | 1.3 | — | kHz |
| | Max. $\lvert f_0 - f_n \rvert_{n=2,\,3,\,4,\,\ldots k}$ | — | 1.5 | — | kHz |
| | Max. $\lvert f_n - f_{n-5} \rvert_{n=6,\,7,\,8,\,\ldots k}$ | — | 0.9 | — | kHz |
| | $\lvert f_1 - f_0 \rvert$ | — | 0.6 | — | kHz |
| Modulation characteristics | $\Delta\, F1_{avg}$ | — | 249.9 | — | kHz |
| | Min. $\Delta\, F2_{max}$ (for at least 99.9% of all $\Delta\, F2_{max}$) | — | 212.1 | — | kHz |
| | $\Delta\, F2_{avg}/\Delta\, F1_{avg}$ | — | 0.88 | — | — |
| In-band emissions | ± 2 MHz offset | — | –29 | — | dBm |
| | ± 3 MHz offset | — | –36 | — | dBm |
| | > ± 3 MHz offset | — | –39 | — | dBm |

Table 6-9. Bluetooth LE - Transmitter Characteristics - 2 Mbps

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Carrier frequency offset and drift | Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$ | — | 2.2 | — | kHz |
| | Max. $\lvert f_0 - f_n \rvert_{n=2,\,3,\,4,\,\ldots k}$ | — | 1.1 | — | kHz |
| | Max. $\lvert f_n - f_{n-5} \rvert_{n=6,\,7,\,8,\,\ldots k}$ | — | 1.1 | — | kHz |
| | $\lvert f_1 - f_0 \rvert$ | — | 0.5 | — | kHz |
| Modulation characteristics | $\Delta\, F1_{avg}$ | — | 499.4 | — | kHz |

Cont'd on next page

Table 6-9 – cont'd from previous page

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| | Min. $\Delta F2_{max}$ (for at least 99.9% of all $\Delta F2_{max}$) | – | 443.5 | – | kHz |
| | $\Delta F2_{avg}/\Delta F1_{avg}$ | – | 0.95 | – | – |
| In-band emissions | ± 4 MHz offset | – | –40 | – | dBm |
| | ± 5 MHz offset | – | –41 | – | dBm |
| | > ± 5 MHz offset | – | –42 | – | dBm |

Table 6-10. Bluetooth LE - Transmitter Characteristics - 125 Kbps

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Carrier frequency offset and drift | Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$ | – | 0.7 | – | kHz |
| | Max. $\lvert f_0 - f_n \rvert_{n=1,\,2,\,3,\,\ldots k}$ | – | 0.3 | – | kHz |
| | $\lvert f_0 - f_3 \rvert$ | – | 0.1 | – | kHz |
| | Max. $\lvert f_n - f_{n-3} \rvert_{n=7,\,8,\,9,\,\ldots k}$ | – | 0.4 | – | kHz |
| Modulation characteristics | $\Delta F1_{avg}$ | – | 250.0 | – | kHz |
| | Min. $\Delta F1_{max}$ (for at least 99.9% of all $\Delta F1_{max}$) | – | 238.0 | – | kHz |
| In-band emissions | ± 2 MHz offset | – | –29 | – | dBm |
| | ± 3 MHz offset | – | –36 | – | dBm |
| | > ± 3 MHz offset | – | –39 | – | dBm |

Table 6-11. Bluetooth LE - Transmitter Characteristics - 500 Kbps

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Carrier frequency offset and drift | Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$ | – | 0.5 | – | kHz |
| | Max. $\lvert f_0 - f_n \rvert_{n=1,\,2,\,3,\,\ldots k}$ | – | 0.3 | – | kHz |
| | $\lvert f_0 - f_3 \rvert$ | – | 0.1 | – | kHz |
| | Max. $\lvert f_n - f_{n-3} \rvert_{n=7,\,8,\,9,\,\ldots k}$ | – | 0.4 | – | kHz |
| Modulation characteristics | $\Delta F2_{avg}$ | – | 230.7 | – | kHz |
| | Min. $\Delta F2_{max}$ (for at least 99.9% of all $\Delta F2_{max}$) | – | 217.6 | – | kHz |
| In-band emissions | ± 2 MHz offset | – | –28 | – | dBm |
| | ± 3 MHz offset | – | –36 | – | dBm |
| | > ± 3 MHz offset | – | –39 | – | dBm |

### 6.2.2 Bluetooth LE RF Receiver (RX) Characteristics

Table 6-12. Bluetooth LE - Receiver Characteristics - 1 Mbps

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Sensitivity @30.8% PER | – | – | –98.5 | – | dBm |
| Maximum received signal @30.8% PER | – | – | 8 | – | dBm |

Cont'd on next page

Table 6-12 – cont'd from previous page

| Parameter | | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| C/I and receiver selectivity performance | Co-channel | F = F0 MHz | — | 7 | — | dB |
| | Adjacent channel | F = F0 + 1 MHz | — | 4 | — | dB |
| | | F = F0 – 1 MHz | — | 3 | — | dB |
| | | F = F0 + 2 MHz | — | –21 | — | dB |
| | | F = F0 – 2 MHz | — | –22 | — | dB |
| | | F = F0 + 3 MHz | — | –28 | — | dB |
| | | F = F0 – 3 MHz | — | –36 | — | dB |
| | | F ≥ F0 + 4 MHz | — | –27 | — | dB |
| | | F ≤ F0 – 4 MHz | — | –36 | — | dB |
| | Image frequency | — | — | –26 | — | dB |
| | Adjacent channel to image frequency | F = $F_{image}$ + 1 MHz | — | –29 | — | dB |
| | | F = $F_{image}$ – 1 MHz | — | –28 | — | dB |
| Out-of-band blocking performance | | 30 MHz ~ 2000 MHz | — | –16 | — | dBm |
| | | 2003 MHz ~ 2399 MHz | — | –24 | — | dBm |
| | | 2484 MHz ~ 2997 MHz | — | –16 | — | dBm |
| | | 3000 MHz ~ 12.75 GHz | — | –1 | — | dBm |
| Intermodulation | | — | — | –27 | — | dBm |

Table 6-13. Bluetooth LE - Receiver Characteristics - 2 Mbps

| Parameter | | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Sensitivity @30.8% PER | | — | — | –95.5 | — | dBm |
| Maximum received signal @30.8% PER | | — | — | 8 | — | dBm |
| C/I and receiver selectivity performance | Co-channel | F = F0 MHz | — | 8 | — | dB |
| | Adjacent channel | F = F0 + 2 MHz | — | 3 | — | dB |
| | | F = F0 – 2 MHz | — | 2 | — | dB |
| | | F = F0 + 4 MHz | — | –23 | — | dB |
| | | F = F0 – 4 MHz | — | –25 | — | dB |
| | | F = F0 + 6 MHz | — | –31 | — | dB |
| | | F = F0 – 6 MHz | — | –35 | — | dB |
| | | F ≥ F0 + 8 MHz | — | –36 | — | dB |
| | | F ≤ F0 – 8 MHz | — | –36 | — | dB |
| | Image frequency | — | — | –23 | — | dB |
| | Adjacent channel to image frequency | F = $F_{image}$ + 2 MHz | — | –30 | — | dB |
| | | F = $F_{image}$ – 2 MHz | — | 3 | — | dB |
| Out-of-band blocking performance | | 30 MHz ~ 2000 MHz | — | –18 | — | dBm |
| | | 2003 MHz ~ 2399 MHz | — | –28 | — | dBm |
| | | 2484 MHz ~ 2997 MHz | — | –16 | — | dBm |
| | | 3000 MHz ~ 12.75 GHz | — | –1 | — | dBm |
| Intermodulation | | — | — | –29 | — | dBm |

**Table 6-14. Bluetooth LE - Receiver Characteristics - 125 Kbps**

| Parameter | | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Sensitivity @30.8% PER | | — | — | –106.0 | — | dBm |
| Maximum received signal @30.8% PER | | — | — | 8 | — | dBm |
| C/I and receiver selectivity performance | Co-channel | F = F0 MHz | — | 2 | — | dB |
| | Adjacent channel | F = F0 + 1 MHz | — | –1 | — | dB |
| | | F = F0 – 1 MHz | — | –3 | — | dB |
| | | F = F0 + 2 MHz | — | –31 | — | dB |
| | | F = F0 – 2 MHz | — | –27 | — | dB |
| | | F = F0 + 3 MHz | — | –33 | — | dB |
| | | F = F0 – 3 MHz | — | –42 | — | dB |
| | | F ≥ F0 + 4 MHz | — | –31 | — | dB |
| | | F ≤ F0 – 4 MHz | — | –48 | — | dB |
| | Image frequency | — | — | –31 | — | dB |
| | Adjacent channel to image frequency | F = $F_{image}$ + 1 MHz | — | –36 | — | dB |
| | | F = $F_{image}$ – 1 MHz | — | –33 | — | dB |

**Table 6-15. Bluetooth LE - Receiver Characteristics - 500 Kbps**

| Parameter | | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Sensitivity @30.8% PER | | — | — | –102.0 | — | dBm |
| Maximum received signal @30.8% PER | | — | — | 8 | — | dBm |
| C/I and receiver selectivity performance | Co-channel | F = F0 MHz | — | 4 | — | dB |
| | Adjacent channel | F = F0 + 1 MHz | — | 1 | — | dB |
| | | F = F0 – 1 MHz | — | –1 | — | dB |
| | | F = F0 + 2 MHz | — | –23 | — | dB |
| | | F = F0 – 2 MHz | — | –24 | — | dB |
| | | F = F0 + 3 MHz | — | –33 | — | dB |
| | | F = F0 – 3 MHz | — | –41 | — | dB |
| | | F ≥ F0 + 4 MHz | — | –31 | — | dB |
| | | F ≤ F0 – 4 MHz | — | –41 | — | dB |
| | Image frequency | — | — | –30 | — | dB |

**Cont'd on next page**

Table 6-15 – cont'd from previous page

| Parameter | | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| | Adjacent channel to image frequency | F = $F_{image}$ + 1 MHz | — | –35 | — | dB |
| | | F = $F_{image}$ – 1 MHz | — | –27 | — | dB |

## 6.3 802.15.4 Radio

Table 6-16. 802.15.4 RF Characteristics

| Name | Description |
|---|---|
| Center frequency range of operating channel | 2405 ~ 2480 MHz |

[1] Zigbee in the 2.4 GHz range supports 16 channels at 5 MHz spacing from channel 11 to channel 26.

### 6.3.1 802.15.4 RF Transmitter (TX) Characteristics

Table 6-17. 802.15.4 Transmitter Characteristics - 250 Kbps

| Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|
| RF transmit power range | –15.0 | — | 20.0 | dBm |
| EVM | — | 13.0% | — | — |

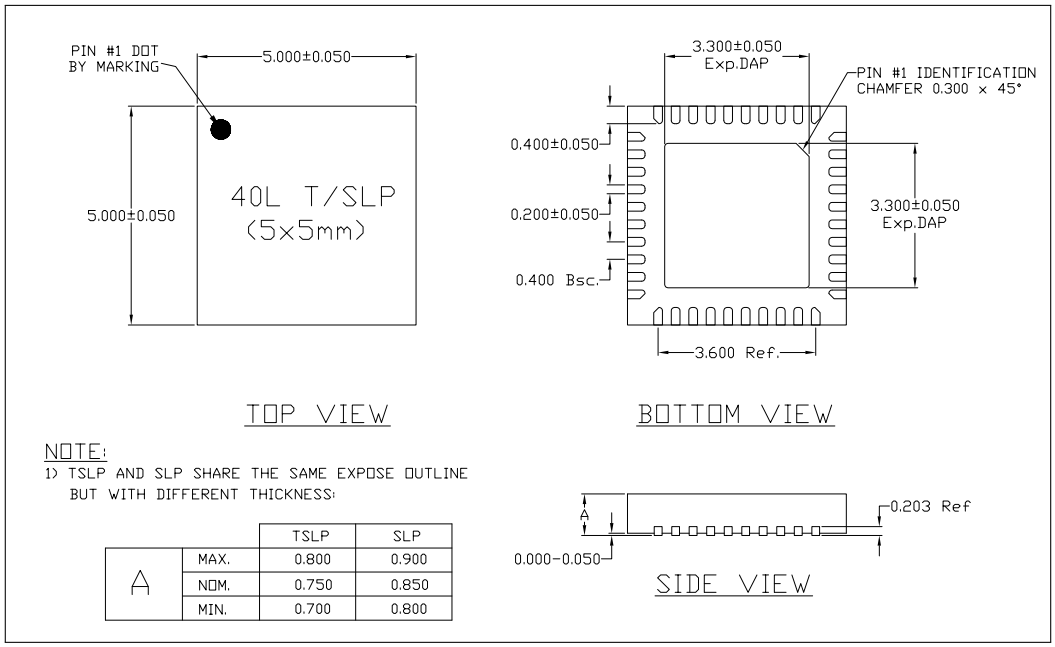
### 6.3.2 802.15.4 RF Receiver (RX) Characteristics

Table 6-18. 802.15.4 Receiver Characteristics - 250 Kbps

| Parameter | | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Sensitivity @1% PER | | — | — | –104.0 | — | dBm |
| Maximum received signal @1% PER | | — | — | 8 | — | dBm |
| Relative jamming level | Adjacent channel | F = F0 + 5 MHz | — | 27 | — | dB |
| | | F = F0 – 5 MHz | — | 32 | — | dB |
| | Alternate channel | F = F0 + 10 MHz | — | 47 | — | dB |
| | | F = F0 – 10 MHz | — | 50 | — | dB |

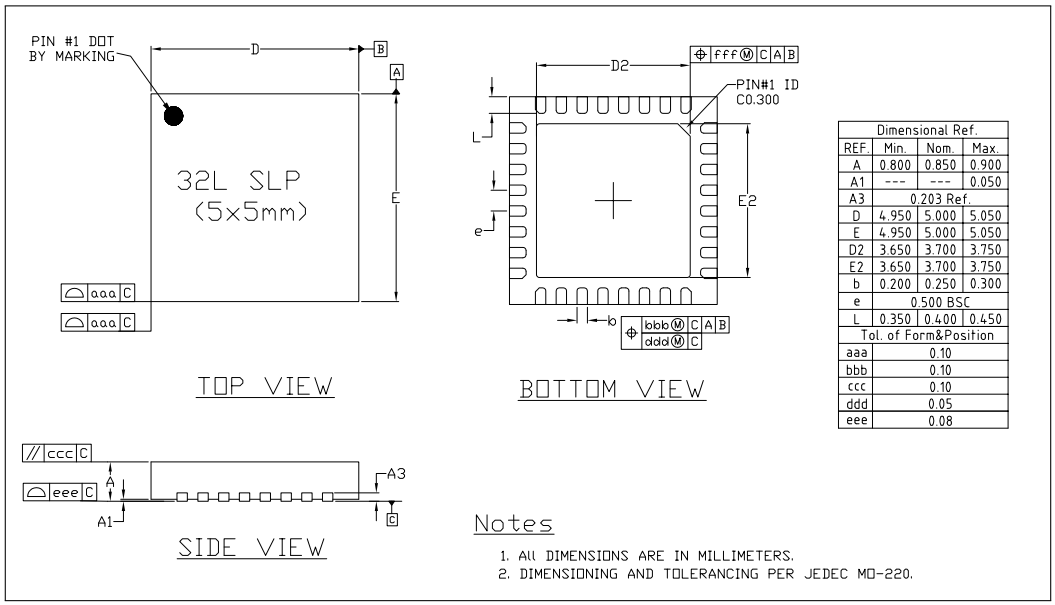
# 7 Packaging

- For information about tape, reel, and chip marking, please refer to *Espressif Chip Packaging Information*.
- The pins of the chip are numbered in anti-clockwise order starting from Pin 1 in the top view. For pin numbers and pin names, see also Figure 2-1 *ESP32-C6 Pin Layout (QFN40, Top View)* and Figure 2-2 *ESP32-C6 Pin Layout (QFN32, Top View)*.
- The recommended land pattern source file (asc) is available for download. You can import the file with software such as PADS and Altium Designer.




| | | TSLP | SLP |
|---|---|---|---|
| A | MAX. | 0.800 | 0.900 |
| | NOM. | 0.750 | 0.850 |
| | MIN. | 0.700 | 0.800 |

**Figure 7-1. QFN40 (5×5 mm) Package**




| Dimensional Ref. | | | |
|---|---|---|---|
| REF. | Min. | Nom. | Max. |
| A | 0.800 | 0.850 | 0.900 |
| A1 | --- | --- | 0.050 |
| A3 | 0.203 Ref. | | |
| D | 4.950 | 5.000 | 5.050 |
| E | 4.950 | 5.000 | 5.050 |
| D2 | 3.650 | 3.700 | 3.750 |
| E2 | 3.650 | 3.700 | 3.750 |
| b | 0.200 | 0.250 | 0.300 |
| e | 0.500 BSC | | |
| L | 0.350 | 0.400 | 0.450 |
| Tol. of Form&Position | | | |
| aaa | 0.10 | | |
| bbb | 0.10 | | |
| ccc | 0.10 | | |
| ddd | 0.05 | | |
| eee | 0.08 | | |

**Figure 7-2. QFN32 (5×5 mm) Package**

# Appendix A – ESP32-C6 Consolidated Pin Overview

Table 7-1. QFN40 Pin Overview

| Pin No. | Pin Name | Pin Type | Pin Providing Power | Pin Settings | | Analog Function | | LP IO MUX Function | | IO MUX Function | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | At Reset | After Reset | 0 | 1 | 0 | 1 | 0 | Type | 1 | Type | 2 | Type |
| 1 | ANT | Analog | | | | | | | | | | | | | |
| 2 | VDDA3P3 | Power | | | | | | | | | | | | | |
| 3 | VDDA3P3 | Power | | | | | | | | | | | | | |
| 4 | CHIP_PU | Analog | | | | | | | | | | | | | |
| 5 | VDDPST1 | Power | | | | | | | | | | | | | |
| 6 | XTAL_32K_P | IO | VDDPST1 | | | XTAL_32K_P | ADC1_CH0 | LP_GPIO0 | LP_UART_DTRN | **GPIO0** | I/O/T | GPIO0 | I/O/T | | |
| 7 | XTAL_32K_N | IO | VDDPST1 | | | XTAL_32K_N | ADC1_CH1 | LP_GPIO1 | LP_UART_DSRN | **GPIO1** | I/O/T | GPIO1 | I/O/T | | |
| 8 | GPIO2 | IO | VDDPST1 | IE | IE | | ADC1_CH2 | LP_GPIO2 | LP_UART_RTSN | **GPIO2** | I/O/T | GPIO2 | I/O/T | FSPIQ | I1/O/T |
| 9 | GPIO3 | IO | VDDPST1 | IE | IE | | ADC1_CH3 | LP_GPIO3 | LP_UART_CTSN | **GPIO3** | I/O/T | GPIO3 | I/O/T | | |
| 10 | MTMS | IO | VDDPST1 | IE | IE | | ADC1_CH4 | LP_GPIO4 | LP_UART_RXD | **MTMS** | I1 | GPIO4 | I/O/T | FSPIHD | I1/O/T |
| 11 | MTDI | IO | VDDPST1 | IE | IE | | ADC1_CH5 | LP_GPIO5 | LP_UART_TXD | **MTDI** | I1 | GPIO5 | I/O/T | FSPIWP | I1/O/T |
| 12 | MTCK | IO | VDDPST1 | | IE, WPU | | ADC1_CH6 | LP_GPIO6 | LP_I2C_SDA | **MTCK** | I1 | GPIO6 | I/O/T | FSPICLK | I1/O/T |
| 13 | MTDO | IO | VDDPST1 | | IE | | | LP_GPIO7 | LP_I2C_SCL | **MTDO** | O/T | GPIO7 | I/O/T | FSPID | I1/O/T |
| 14 | GPIO8 | IO | VDDPST2 | IE | IE | | | | | **GPIO8** | I/O/T | GPIO8 | I/O/T | | |
| 15 | GPIO9 | IO | VDDPST2 | IE, WPU | IE, WPU | | | | | **GPIO9** | I/O/T | GPIO9 | I/O/T | | |
| 16 | GPIO10 | IO | VDDPST2 | | IE | | | | | **GPIO10** | I/O/T | GPIO10 | I/O/T | | |
| 17 | GPIO11 | IO | VDDPST2 | | IE | | | | | **GPIO11** | I/O/T | GPIO11 | I/O/T | | |
| 18 | GPIO12 | IO | VDDPST2 | | IE | **USB_D-** | | | | GPIO12 | I/O/T | GPIO12 | I/O/T | | |
| 19 | GPIO13 | IO | VDDPST2 | | IE, WPU | **USB_D+** | | | | GPIO13 | I/O/T | GPIO13 | I/O/T | | |
| 20 | SPICS0 | IO | VDD_SPI | WPU | IE, WPU | | | | | **SPICS0** | O/T | GPIO24 | I/O/T | | |
| 21 | SPIQ | IO | VDD_SPI | WPU | IE, WPU | | | | | **SPIQ** | I1/O/T | GPIO25 | I/O/T | | |
| 22 | SPIWP | IO | VDD_SPI | WPU | IE, WPU | | | | | SPIWP | I1/O/T | **GPIO26** | I/O/T | | |
| 23 | VDD_SPI | Power/IO | — | | | **VDD_SPI** | | | | GPIO27 | I/O/T | GPIO27 | I/O/T | | |
| 24 | SPIHD | IO | VDD_SPI | WPU | IE, WPU | | | | | SPIHD | I1/O/T | **GPIO28** | I/O/T | | |
| 25 | SPICLK | IO | VDD_SPI | WPU | IE, WPU | | | | | **SPICLK** | O/T | GPIO29 | I/O/T | | |
| 26 | SPID | IO | VDD_SPI | WPU | IE, WPU | | | | | **SPID** | I1/O/T | GPIO30 | I/O/T | | |
| 27 | GPIO15 | IO | VDDPST2 | IE | IE | | | | | **GPIO15** | I/O/T | GPIO15 | I/O/T | | |
| 28 | VDDPST2 | Power | | | | | | | | | | | | | |
| 29 | U0TXD | IO | VDDPST2 | | WPU | | | | | **U0TXD** | O | GPIO16 | I/O/T | FSPICS0 | I1/O/T |
| 30 | U0RXD | IO | VDDPST2 | | IE, WPU | | | | | **U0RXD** | I1 | GPIO17 | I/O/T | FSPICS1 | O/T |
| 31 | SDIO_CMD | IO | VDDPST2 | WPU | IE | | | | | **SDIO_CMD** | I1/O/T | GPIO18 | I/O/T | FSPICS2 | O/T |
| 32 | SDIO_CLK | IO | VDDPST2 | WPU | IE | | | | | **SDIO_CLK** | I1 | GPIO19 | I/O/T | FSPICS3 | O/T |
| 33 | SDIO_DATA0 | IO | VDDPST2 | WPU | IE | | | | | **SDIO_DATA0** | I1/O/T | GPIO20 | I/O/T | FSPICS4 | O/T |
| 34 | SDIO_DATA1 | IO | VDDPST2 | WPU | IE | | | | | **SDIO_DATA1** | I1/O/T | GPIO21 | I/O/T | FSPICS5 | O/T |
| 35 | SDIO_DATA2 | IO | VDDPST2 | WPU | IE | | | | | **SDIO_DATA2** | I1/O/T | GPIO22 | I/O/T | | |
| 36 | SDIO_DATA3 | IO | VDDPST2 | WPU | IE | | | | | **SDIO_DATA3** | I1/O/T | GPIO23 | I/O/T | | |
| 37 | VDDA1 | Power | | | | | | | | | | | | | |
| 38 | XTAL_N | Analog | | | | | | | | | | | | | |
| 39 | XTAL_P | Analog | | | | | | | | | | | | | |
| 40 | VDDA2 | Power | | | | | | | | | | | | | |
| 41 | GND | Power | | | | | | | | | | | | | |

* For details, see Section 2 *Pins*. Regarding highlighted cells, see Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*.

## Table 7-2. QFN32 Pin Overview

| Pin No. | Pin Name | Pin Type | Pin Providing Power | Pin Settings | | Analog Function | | LP IO MUX Function | | IO MUX Function | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | At Reset | After Reset | 0 | 1 | 0 | 1 | 0 | Type | 1 | Type | 2 | Type |
| 1 | ANT | Analog | | | | | | | | | | | | | |
| 2 | VDDA3P3 | Power | | | | | | | | | | | | | |
| 3 | VDDA3P3 | Power | | | | | | | | | | | | | |
| 4 | CHIP_PU | Analog | | | | | | | | | | | | | |
| 5 | VDDPST1 | Power | | | | | | | | | | | | | |
| 6 | XTAL_32K_P | IO | VDDPST1 | | | XTAL_32K_P | ADC1_CH0 | LP_GPIO0 | LP_UART_DTRN | **GPIO0** | I/O/T | GPIO0 | I/O/T | | |
| 7 | XTAL_32K_N | IO | VDDPST1 | | | XTAL_32K_N | ADC1_CH1 | LP_GPIO1 | LP_UART_DSRN | **GPIO1** | I/O/T | GPIO1 | I/O/T | | |
| 8 | GPIO2 | IO | VDDPST1 | IE | IE | | ADC1_CH2 | LP_GPIO2 | LP_UART_RTSN | **GPIO2** | I/O/T | GPIO2 | I/O/T | FSPIQ | I1/O/T |
| 9 | GPIO3 | IO | VDDPST1 | IE | IE | | ADC1_CH3 | LP_GPIO3 | LP_UART_CTSN | **GPIO3** | I/O/T | GPIO3 | I/O/T | | |
| 10 | MTMS | IO | VDDPST1 | IE | IE | | ADC1_CH4 | LP_GPIO4 | LP_UART_RXD | **MTMS** | I1 | GPIO4 | I/O/T | FSPIHD | I1/O/T |
| 11 | MTDI | IO | VDDPST1 | IE | IE | | ADC1_CH5 | LP_GPIO5 | LP_UART_TXD | **MTDI** | I1 | GPIO5 | I/O/T | FSPIWP | I1/O/T |
| 12 | MTCK | IO | VDDPST1 | | IE, WPU | | ADC1_CH6 | LP_GPIO6 | LP_I2C_SDA | **MTCK** | I1 | GPIO6 | I/O/T | FSPICLK | I1/O/T |
| 13 | MTDO | IO | VDDPST1 | | IE | | | LP_GPIO7 | LP_I2C_SCL | **MTDO** | O/T | GPIO7 | I/O/T | FSPID | I1/O/T |
| 14 | GPIO8 | IO | VDDPST2 | IE | IE | | | | | **GPIO8** | I/O/T | GPIO8 | I/O/T | | |
| 15 | GPIO9 | IO | VDDPST2 | IE, WPU | IE, WPU | | | | | **GPIO9** | I/O/T | GPIO9 | I/O/T | | |
| 16 | GPIO12 | IO | VDDPST2 | | IE | **USB_D-** | | | | GPIO12 | I/O/T | GPIO12 | I/O/T | | |
| 17 | GPIO13 | IO | VDDPST2 | | IE, WPU | **USB_D+** | | | | GPIO13 | I/O/T | GPIO13 | I/O/T | | |
| 18 | GPIO14 | IO | VDDPST2 | | IE | | | | | **GPIO14** | I/O/T | GPIO14 | I/O/T | | |
| 19 | GPIO15 | IO | VDDPST2 | IE | IE | | | | | **GPIO15** | I/O/T | GPIO15 | I/O/T | | |
| 20 | VDDPST2 | Power | | | | | | | | | | | | | |
| 21 | U0TXD | IO | VDDPST2 | | WPU | | | | | **U0TXD** | O | GPIO16 | I/O/T | FSPICS0 | I1/O/T |
| 22 | U0RXD | IO | VDDPST2 | | IE, WPU | | | | | **U0RXD** | I1 | GPIO17 | I/O/T | FSPICS1 | O/T |
| 23 | SDIO_CMD | IO | VDDPST2 | WPU | IE | | | | | **SDIO_CMD** | I1/O/T | GPIO18 | I/O/T | FSPICS2 | O/T |
| 24 | SDIO_CLK | IO | VDDPST2 | WPU | IE | | | | | **SDIO_CLK** | I1 | GPIO19 | I/O/T | FSPICS3 | O/T |
| 25 | SDIO_DATA0 | IO | VDDPST2 | WPU | IE | | | | | **SDIO_DATA0** | I1/O/T | GPIO20 | I/O/T | FSPICS4 | O/T |
| 26 | SDIO_DATA1 | IO | VDDPST2 | WPU | IE | | | | | **SDIO_DATA1** | I1/O/T | GPIO21 | I/O/T | FSPICS5 | O/T |
| 27 | SDIO_DATA2 | IO | VDDPST2 | WPU | IE | | | | | **SDIO_DATA2** | I1/O/T | GPIO22 | I/O/T | | |
| 28 | SDIO_DATA3 | IO | VDDPST2 | WPU | IE | | | | | **SDIO_DATA3** | I1/O/T | GPIO23 | I/O/T | | |
| 29 | VDDA1 | Power | | | | | | | | | | | | | |
| 30 | XTAL_N | Analog | | | | | | | | | | | | | |
| 31 | XTAL_P | Analog | | | | | | | | | | | | | |
| 32 | VDDA2 | Power | | | | | | | | | | | | | |
| 33 | GND | Power | | | | | | | | | | | | | |

* For details, see Section 2 *Pins*. Regarding highlighted cells, see Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*.

# Glossary

| | | |
|---|---|---|
| **eFuse** | A one-time programmable (OTP) memory which stores system and user parameters, such as MAC address, chip revision number, flash encryption key, etc. Value 0 indicates the default state, and value 1 indicates the eFuse has been programmed | 36 |
| **eFuse parameter** | A parameter stored in an electrically programmable fuse (eFuse) memory within a chip. The parameter can be set by programming EFUSE_PGM_DATA*n*_REG registers, and read by reading a register field named after the parameter | 29 |
| **in-package flash** | Flash integrated directly into the chip's package, and external to the chip die | 4, 28 |
| **joint download boot mode** | A boot mode in which users can download code into flash via the UART or other interfaces (see Table 3-3 *Chip Boot Mode Control* > Note), and load and execute the downloaded code from the flash or SRAM | 30 |
| **module** | A self-contained unit integrated within the chip to extend its capabilities, such as cryptographic modules, RF modules | 2 |
| **off-package flash** | Flash external to the chip's package | 28, 36 |
| **peripheral** | A hardware component or subsystem within the chip to interface with the outside world | 2 |
| **SPI boot mode** | A boot mode in which users load and execute the existing code from SPI flash | 30 |
| **strapping pin** | A type of GPIO pin used to configure certain operational settings during the chip's power-up, and can be reconfigured as normal GPIO after the chip's reset | 29 |

# Related Documentation and Resources

## Related Documentation

- ESP32-C6 Technical Reference Manual – Detailed information on how to use the ESP32-C6 memory and peripherals.
- ESP32-C6 Hardware Design Guidelines – Guidelines on how to integrate the ESP32-C6 into your hardware product.
- *Certificates*
  https://espressif.com/en/support/documents/certificates
- *ESP32-C6 Product/Process Change Notifications (PCN)*
  https://espressif.com/en/support/documents/pcns?keys=ESP32-C6
- *Documentation Updates and Update Notification Subscription*
  https://espressif.com/en/support/download/documents

## Developer Zone

- ESP-IDF Programming Guide for ESP32-C6 – Extensive documentation for the ESP-IDF development framework.
- *ESP-IDF* and other development frameworks on GitHub.
  https://github.com/espressif
- *ESP32 BBS Forum* – Engineer-to-Engineer (E2E) Community for Espressif products where you can post questions, share knowledge, explore ideas, and help solve problems with fellow engineers.
  https://esp32.com/
- *The ESP Journal* – Best Practices, Articles, and Notes from Espressif folks.
  https://blog.espressif.com/
- See the tabs *SDKs and Demos*, *Apps*, *Tools*, *AT Firmware*.
  https://espressif.com/en/support/download/sdks-demos

## Products

- *ESP32-C6 Series SoCs* – Browse through all ESP32-C6 SoCs.
  https://espressif.com/en/products/socs?id=ESP32-C6
- *ESP32-C6 Series Modules* – Browse through all ESP32-C6-based modules.
  https://espressif.com/en/products/modules?id=ESP32-C6
- *ESP32-C6 Series DevKits* – Browse through all ESP32-C6-based devkits.
  https://espressif.com/en/products/devkits?id=ESP32-C6
- *ESP Product Selector* – Find an Espressif hardware product suitable for your needs by comparing or applying filters.
  https://products.espressif.com/#/product-selector?language=en

## Contact Us

- See the tabs *Sales Questions*, *Technical Enquiries*, *Circuit Schematic & PCB Design Review*, *Get Samples* (Online stores), *Become Our Supplier*, *Comments & Suggestions*.
  https://espressif.com/en/contact-us/sales-questions

# Revision History

| Date | Version | Release notes |
|---|---|---|
| 2024-05-10 | v1.1 | • Updated CPU CoreMark® scode in Section Product Overview<br>• Added flash erase cycles, retention time, maximum clock frequency in Section 4.1.2.1 *Internal Memory*<br>• Updated cumulative IO output current in Table 5-1 *Absolute Maximum Ratings*<br>• Updated the value of $R_{SPI}$ in Table 5-3 *VDD_SPI Internal and Output Characteristics*<br>• Added links and descriptions of PCB land pattern in Section 7 *Packaging*<br>• Added Section Glossary<br>• Improved the formatting, structure, and wording in the following sections:<br>– Section 2 *Pins*<br>– Section 3 *Boot Configurations* (used to be named as "Strapping Pins")<br>– Section 4 *Functional Description*<br>• Other minor updates |
| 2023-07-25 | v1.0 | • Added descriptions of USB_PU in Table 2-4 *QFN40 IO MUX Pin Functions* and Table 2-5 *QFN32 IO MUX Pin Functions*, note 4<br>• Updated Section 3.3 *ROM Messages Printing Control*<br>• Added Section 5.5 *ADC Characteristics*<br>• Updated the measurement conditions in Table 5-8 *Current Consumption for Bluetooth LE in Active Mode* and Table 5-9 *Current Consumption for 802.15.4 in Active Mode* from –24.0 dBm to –15.0 dBm, and the corresponding peak values<br>• Added Section 5.7 *Reliability*<br>• Updated the minimum value of RF transmit power range to –15.0 dBm in Table 6-7 *Bluetooth LE RF Characteristics* and Table 6-17 *802.15.4 Transmitter Characteristics - 250 Kbps*<br>• Updated Related Documentation and Resources<br>• Other minor changes |
| 2023-01-16 | v0.5 | Preliminary release |

www.espressif.com

## Disclaimer and Copyright Notice